

Trabajo de Fin de Grado

Grado de Ingeniería en Tecnologías Industriales

**Desarrollo de una red neuronal para la predicción de la
producción energética de un aerogenerador**

MEMORIA

Autor: Miguel Hernández Mestre
Director: Ricard Ferrer San José
Convocatoria: Junio 2020



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resumen

Este proyecto está enmarcado en el ámbito de la ciencia de datos y detalla el proceso de desarrollo y entrenamiento de un sistema informático capaz de pronosticar, con la mayor exactitud posible, la producción de energía eléctrica y retribución económica de un aerogenerador a corto plazo.

El objeto principal de este proyecto es programar en lenguaje *Python* un algoritmo basado en el aprendizaje profundo, que mediante la utilización de una red neuronal de memoria a corto y largo plazo (LSTM), sea capaz de aprender de experiencias del pasado y generar los pronósticos futuros mencionados anteriormente. Estas predicciones pretenden ser lo más precisas posibles, y para ello, se debe extraer el máximo aprovechamiento de los datos disponibles realizando diferentes modificaciones y análisis sobre ellos y sobre el algoritmo.

Las acciones que se llevan a cabo para la consecución de los objetivos propuestos se agrupan en tres fases principales:

- En primer lugar, se realiza un análisis exploratorio de datos (EDA) que permite un estudio profundo y una adecuación de los datos al problema que se enfrenta.
- A continuación, se construye el sistema informático o modelo. Para ello se ejecuta un test de desarrollo que permite escoger la estructura que maximiza la precisión de los pronósticos.
- Finalmente, el entrenamiento del modelo permite la creación del algoritmo computacional.

El resultado obtenido enseña como una correcta gestión de los datos y un exhaustivo análisis sobre todas las variables estructurales que permite la red LSTM da la opción de crear un modelo pronosticador con un error aceptable. Además de una metodología capaz de desarrollar y entrenar una red neuronal.

Sumario

SUMARIO	5
1. GLOSARIO	9
2. PREFACIO	11
2.1. Motivación	11
2.2. Requerimientos previos.....	11
3. INTRODUCCIÓN	13
3.1. Objetivos del proyecto	13
3.2. Alcance del proyecto	14
4. LA ENERGÍA EÓLICA	15
4.1. El recurso eólico	15
4.2. La ciencia de datos en la energía eólica	16
4.3. Mercado eléctrico eólico español	16
5. REDES NEURONALES ARTIFICIALES (RNA)	18
5.1. Origen.....	19
5.2. Redes neuronales recurrentes (RNR).....	19
5.2.1. Redes de memoria a corto y largo plazo (LSTM).....	20
5.3. Proceso de aprendizaje.....	21
5.3.1. Función de activación	22
5.3.2. Función de pérdida	23
5.3.3. Propagación hacia atrás	23
5.3.4. Sobreajuste y sub-ajuste.....	23
6. ANÁLISIS EXPLORATORIO DE DATOS	25
6.1. Marco de datos.....	25
6.2. Análisis objetivo.....	25
6.3. Valores atípicos	29
6.4. Eliminar datos faltantes	33
6.5. Correlaciones	34
6.5.1. Estudio de correlación en el modelo frecuencial 1	35
6.5.2. Estudio de correlación en el modelo frecuencial 2	36
6.6. Estacionalidad	38
6.6.1. Estacionalidad trimestral	39
6.6.2. Estacionalidad mensual	40
6.6.3. Estacionalidad diaria	43

7. PROCESO DE CONSTRUCCIÓN	45
7.1. La arquitectura de la red	46
7.2. Normalización de los datos	47
7.3. Funciones de activación	47
7.4. Algoritmos de optimización del aprendizaje.....	49
7.5. Número de épocas y tamaño del lote	51
7.6. Creación de sub-modelos	51
7.6.1. Variaciones basadas en el modelo frecuencial 1	52
7.6.2. Variaciones basadas en el modelo frecuencial 2	53
7.6.3. Transformación en aprendizaje supervisado	53
7.7. Test de desarrollo	55
7.7.1. Indicador clave de rendimiento (KPI)	55
7.7.2. Pasos.....	56
7.7.2.1. Paso 1: Algoritmo optimizador y funciones de activación.....	56
7.7.2.2. Paso 2: Sub-modelos.....	60
7.7.2.3. Paso 3: Número de neuronas de la HL	61
7.7.2.4. Paso 4: Medida del lote.....	64
7.7.2.5. Paso 5: Número de épocas.....	66
7.7.3. Resultados.....	68
8. EL MODELO PREDICTIVO	69
8.1. Entrenamiento de la red.....	69
8.1.1. División en entrenamiento y validación	69
8.1.2. Ejecución del entrenamiento y resultados obtenidos	71
8.2. Análisis de las predicciones.....	73
8.2.1. Visión general	73
8.2.2. Pronosticación de la retribución económica	75
8.2.3. Valoración de las predicciones	77
9. ANÁLISIS DE COSTES	80
9.1. Coste del trabajo humano.....	80
9.2. Coste del hardware y software	80
9.3. Coste total.....	81
10. IMPACTO AMBIENTAL	82
CONCLUSIONES	83
AGRADECIMIENTOS	85

BIBLIOGRAFÍA	86
Referencias bibliográficas	86
Bibliografía complementaria	87

1. Glosario

- **Aprendizaje supervisado:** Es la tarea del aprendizaje automático de aprender una función que asigna una entrada a una salida basándose en ejemplos de pares de entrada-salida.
- **EDA** (Análisis Exploratorio de Datos): Es una visión general sobre conjuntos de datos, para resumir sus características principales, a menudo con métodos visuales.
- **ERM** (Error Relativo Medio): Es una medida de error comparativa frecuente. Su fórmula es la media del sumatorio de la diferencia entre el valor real.
- **GD** (Descenso de Gradiente): Es un proceso matemático iterativo de primer orden para encontrar un mínimo local de una función diferenciable.
- **GPUs** (Unidad de Procesamiento Gráfico): Es un procesador especial utilizado en hardware de gráficos para ordenadores. Capaz de realizar gran cantidad de cálculos.
- **Hiperparámetros:** En aprendizaje automático, es un parámetro cuyo valor se utiliza para controlar el proceso de aprendizaje.
- **IEC II:** Clasificación de los aerogeneradores según las características del viento. Esta clase se denomina de viento medio, con una velocidad media anual de 8,5m/s.
- **IL, OL, HL** (Capa de Entrada, Capa de Salida y Capa Oculta): En redes neuronales, las capas son conjuntos de neuronas y pueden ser de tres tipos, de entrada, de salida y ocultas.
- **LSTM** (Memoria a Corto y Largo plazo): Es un tipo de RNR que puede memorizar dependencias a largo y corto plazo entre los elementos de una secuencia.
- **MAE** (Error Medio Absoluto): Es una medida de error frecuente cuya fórmula es la media de la suma de las diferencias.
- **OMIE** (Operador del Mercado Ibérico de Energía): Es el organismo operador de todos los mercados de la electricidad de la Península Ibérica.
- **RNA** (Redes Neuronales Artificiales): Es un modelo computacional que consta de varios elementos de procesamiento que reciben entradas y entregan salidas basadas en sus funciones de activación predefinidas.

- **RNR** (Redes Neuronales Recurrentes): Es un tipo de RNA que incluye bucles de realimentación que permiten que la información perdure un cierto periodo de tiempo durante la computación.
- **SCADA** (Supervisión, Control y Adquisición de Datos): Es un sistema software que supervisa procesos desde lejos gracias a la adquisición de datos de procesos remotos.
- **Sesgo**: Es una constante que ayuda al modelo a ajustarse mejor a los datos.
- **TME** (Tiempo medio por Época): Es el tiempo que tarda el modelo en computar una época, es decir, recorrer todo el conjunto de muestras.
- **Weibull**: Es una distribución de probabilidad continua ampliamente utilizada en análisis de datos de vida debido a su versatilidad.

2. Prefacio

Durante la realización de las prácticas curriculares en un puesto de análisis de datos se presentó la necesidad de realizar el Trabajo de Fin de Grado. El interés que siempre habían causado en mí las energías renovables me hizo buscar entre las ofertas de la bolsa de trabajos de la ETSEIB una relacionada con este tema. Después de diversas reuniones con el tutor se decidió juntar el campo de las energías renovables, en especial la eólica, y el de la ciencia y el análisis de datos, en el cual se estaba familiarizado.

2.1. Motivación

Hoy en día, uno de los problemas más importante que sufre la humanidad y el lugar que habitamos es la contaminación y su impacto en el medio ambiente, esta contaminación está producida en gran medida por culpa de la producción de energía eléctrica [8]. Además, la producción y el consumo de energía eléctrica es uno de los campos más afectados por la tecnología, ya que casi todo funciona con electricidad. Por eso, fomentar las energías renovables y en este caso una que aprovecha algo tan común y accesible como el viento es fundamental. Poder predecir la producción de energía con mayor exactitud mejora la capacidad de la red de distribución y su rendimiento evitando las posibles fluctuaciones y sus consecuencias, así como una mejor participación en los mercados mayoristas de energía eléctrica. Utilizar la ciencia de datos ayuda a realizar modelos con más exactitud, basados en muestras previas reales, que en estos casos personalizan las producciones para cada turbina. Todo esto desemboca en una reducción de costes y reducción de la huella en el medio ambiente.

2.2. Requerimientos previos

El proyecto requiere de capacidad de análisis de datos a partir de los cuales extraer conclusiones, así como de la utilización de métodos estadísticos para la toma de decisiones, para los cuales se ha utilizado el programa Minitab. Además, también resulta necesario tener conocimientos de programación en *Python* para poder crear el modelo predictivo. Algunos de estos conocimientos los aporta el Grado en Tecnologías Industriales.

Por otra parte, también es cierto que algunos de los puntos que se tratan en el trabajo son complementarios al grado ya que la creación de sistemas predictivos es un tema que no se trata en el grado. Sin embargo, estos sí que se basan en conocimientos adquiridos durante dicho grado y otros aprendidos durante el periodo de prácticas en la empresa. Por esto mencionado y por la motivación que este tema despierta en mí de cara a futuras formaciones

académicas, resulta de especial interés ampliar conocimientos en este ámbito.

3. Introducción

Este proyecto pretende predecir la producción de energía eléctrica y la retribución de una turbina eólica en particular para un período de 24 horas vista logrando una cierta precisión. Esta referencia temporal es seleccionada con relación al precio de la energía en el mercado diario español. Para los países de la Unión Europea, como es el caso de España, la subasta tiene lugar una vez al día; a las 12 p.m., donde se determina el precio de la energía eléctrica de las 24 horas del día siguiente [4].

La predicción se realiza con un algoritmo de aprendizaje automático de la clase aprendizaje profundo. Este algoritmo de aprendizaje supervisado contiene una red neuronal artificial del tipo LSTM, utilizada normalmente en traducción de texto y reconocimiento de escritura a mano, que intenta imitar las conexiones cerebrales.

Los datos utilizados provienen de un parque eólico en el sur de España. Contienen las muestras recogidas por un sistema SCADA de distintas señales de 4 aerogeneradores diferentes, de un total de 12 instalados en el parque eólico, correspondientes a los años 2016 y 2017. El intervalo entre muestra y muestra es de 10 minutos, esto y la variedad de muestras de la que se dispone da la posibilidad de desarrollar distintos modelos que se compararán entre ellos.

3.1. Objetivos del proyecto

El objetivo principal de este proyecto consiste en crear un modelo predictivo capaz de predecir con la mayor exactitud posible la producción energética y retribución económica de un aerogenerador a 24 horas vista a partir de:

- Realizar un análisis exploratorio de los datos disponibles para sacar el mayor provecho posible de estos.
- Construir un modelo predictivo que minimice el error de salida.

Esta segunda acción debe conseguirse mediante:

- Programar con lenguaje *Python*.
- Utilizar la tecnología de aprendizaje profundo y en especial una red neuronal recurrente del tipo LSTM.
- Comparar diferentes modelos y obtener el óptimo realizando un test de

desarrollo.

- Analizar los pronósticos y valorar el nivel de precisión conseguido.

3.2. Alcance del proyecto

Como se ha comentado anteriormente, el proyecto consiste en la creación de un modelo predictivo basado en redes neuronales del tipo LSTM capaz de predecir la producción energética a corto plazo con la mayor precisión posible. Por lo tanto, el proyecto finaliza con el desarrollo de este modelo con una metodología ideada según la investigación realizada previamente. Además, se analizarán los pronósticos generados por este modelo y su aplicación a las predicciones sobre la retribución económica.

El campo de las redes neuronales y en especial el aprendizaje automático es inmenso y se encuentra en constante desarrollo. Por este motivo, se deja para una posible segunda fase de ampliación del presente proyecto la creación de distintos tipos de pronosticadores.

4. La energía eólica

La energía eólica es la que se obtiene del aprovechamiento de la energía cinética de las masas de aire, es decir, del viento. En el pasado movía barcos y molinos, en la actualidad, provee de electricidad limpia e inagotable. Para el aprovechamiento de este recurso renovable se utilizan los aerogeneradores.

España ha incrementado la potencia eólica en 2.243 megavatios en 2019, multiplicando por cinco la potencia instalada respecto al año anterior, sumando un total de 25.704 megavatios eólicos instalados. La eólica ha producido un 20,8% de la electricidad consumida en 2019 [1]. Gracias a esto, España se ha convertido en una potencia a nivel europeo e incluso a nivel mundial y un gran contribuyente a las energías limpias.

En este proyecto, los datos utilizados provienen de 4 aerogeneradores del modelo *V90/2000* con una potencia nominal de 2 megavatios fabricados por la compañía danesa *Vestas*.

4.1. El recurso eólico

El origen del viento está relacionado directamente con dos factores. Las diferencias de temperatura entre los polos y el ecuador causan movimientos de aire a escala global, también llamados vientos globales. A nivel local, la diferencia de materiales en la superficie terrestre, tierra y agua, genera movimientos de aire provocados por la diferencia de temperatura entre el día y la noche.

La variabilidad del viento es un obstáculo con el que hay que lidiar siempre que se realicen predicciones donde se contenga esta variable. Esta variabilidad puede ser a corto plazo, como son las ráfagas o los cambios de dirección, o a largo plazo, como es el caso de las diferencias entre el día y la noche e incluso las estaciones del año. Por eso es importante entender la variabilidad también de los datos del viento y diferenciar cada caso para saber que valores son útiles y cuales irrelevantes en un proyecto como este.

Otros aspectos relevantes que hacen al viento más impredecible y difícil de parametrizar son los obstáculos artificiales, la topografía y la rugosidad del terreno de la zona de estudio. Estos aspectos y los mencionados con anterioridad obligan a clasificar el viento por clases de viento y turbulencia. Esta clasificación permite crear también aerogeneradores que saquen el máximo rendimiento a cada una de estas clases. Los aerogeneradores que proporcionan los datos para este proyecto son de la clase de viento del tipo *IEC II*.

4.2. La ciencia de datos en la energía eólica

Los pronósticos de viento es algo muy común y necesitado para muchas actividades diferentes hoy en día, y se vienen utilizando desde mediados del siglo pasado [2]. Actualmente, gracias a la ciencia de datos y la mejora en las capacidades computacionales, las predicciones han dado un salto cualitativo. La energía eólica depende directamente del viento, como su nombre indica, por eso una predicción del viento lo más precisa posible puede dar lugar a conocer la producción de energía con un error mínimo. Los modelos estadísticos predominan en las predicciones de este tipo.

Otro campo con mucho auge dentro de la energía eólica es el mantenimiento predictivo. Los aerogeneradores son máquinas de grandes dimensiones creadas por el hombre que cuestan miles de euros, evitar reparaciones innecesarias pueden llegar a suponer ahorros de cinco millones de euros para un parque eólico [3]. Una buena gestión de los datos puede adelantar el retorno de la inversión en un parque eólico reduciendo costes innecesarios.

Por último, las tecnologías de ciencia de datos más avanzadas como las utilizadas en este proyecto permiten predecir todo tipo de señales de un aerogenerador, incluso sin necesidad de que este exista en esa ubicación realmente. Esto permite identificar zonas óptimas donde construir parques eólicos sin necesidad de riesgo de inversión. Además de predecir las producciones futuras con únicamente parámetros determinados característicos de esa zona. Esta tecnología se utiliza en este proyecto.

4.3. Mercado eléctrico eólico español

El mercado eléctrico español es un campo de lo más complejo y digno de estudio, aun así, en este proyecto se obviarán muchos de los aspectos que lo hacen tan especial. Aun así, en este apartado se cita brevemente su funcionamiento y qué papel tiene la energía eólica dentro de este, siempre desde el punto de vista de los productores.

Para empezar, el precio de la energía eléctrica de las 24 horas de un día queda determinado íntegramente en una subasta a las 12 p.m. del día anterior. El Operador del Mercado Ibérico de Energía (OMIE), que se encarga de organizar todo el mercado eléctrico, analiza todas las ofertas/precios de las compañías generadoras de energía eléctrica más baratas. Esto junto a los datos de previsión de la demanda aportados por las comercializadoras también a esta organización dan lugar al precio final del megavatio-hora, uno para cada hora del total de 24 que tiene un día. Lo que hace especial a este mercado es que el precio de la energía eléctrica

es el de la última oferta de las empresas generadoras de luz con las que se haya cubierto la demanda de las comercializadoras. Es decir, la oferta más cara es la que acaba estableciéndose como precio a pagar por la electricidad, este concepto se llama mercado marginalista. Las unidades del precio de la energía son los euros por megavatio hora (€/MWh). Por esta misma razón, una previsión lo más precisa posible puede reducir costes a todos los usuarios.

Por otro lado, la energía eólica, aparte de ser una energía renovable también es barata desde el punto de vista del coste de producción. Esta necesita poco mantenimiento, además de utilizar un recurso gratis como es el viento, por este motivo su coste marginal es bajo. El orden de mérito es la forma de clasificar las fuentes de energía disponibles, en función del coste marginal ascendente. Todo esto hace que las renovables, y en este caso la eólica se consuma como primera opción y evitando, de una manera involuntaria, la generación con combustibles. La continua aportación de energía a la red permite que los productores obtengan su máxima retribución de manera continua, pero también se ve perjudicada por la mediocridad de sus precios.

En último lugar, hay que mencionar que existe un mercado intradiario que permite comprar y vender electricidad, incluso a nivel europeo. Este campo es complejo y no se tendrá en cuenta para este proyecto [4].

5. Redes neuronales artificiales (RNA)

Una red neuronal artificial es un paradigma de procesamiento de información inspirado en el cerebro humano. Es una simulación biológica ejecutada en un ordenador para realizar ciertas tareas específicas como la agrupación, clasificación y reconocimiento de patrones. Inspirados por el cerebro humano, las RNA pueden aprender y generar resultados de la experiencia. El área principal de aplicación de las RNA es la predicción por diversas razones: la primera, aprenden de ejemplos y capturan sutiles relaciones entre datos; la segunda, las redes neuronales pueden generalizar, infiriendo en la parte invisible de los datos, predicciones del comportamiento futuro de la población; la tercera, las RNA son aproximadores funcionales universales, tienen formas más generales y flexibles que los métodos estadísticos tradicionales; la última, son capaces de realizar modelos no lineales sin un previo conocimiento sobre las relaciones entre las variables de entrada y salida [9].

Las RNA son sistemas de aprendizaje supervisado formado por una gran cantidad de neuronas o perceptrones. Cada unidad básica puede tomar decisiones propias y transmitir esas decisiones a otras neuronas, organizadas en capas interconectadas. Cada una de estas conexiones entre neuronas tiene un peso asociado, que representa la importancia de la correspondiente conexión al resultado final. Una red básica tiene: una capa de entrada, una o varias capas ocultas y una capa de salida (ver Figura.5.1).

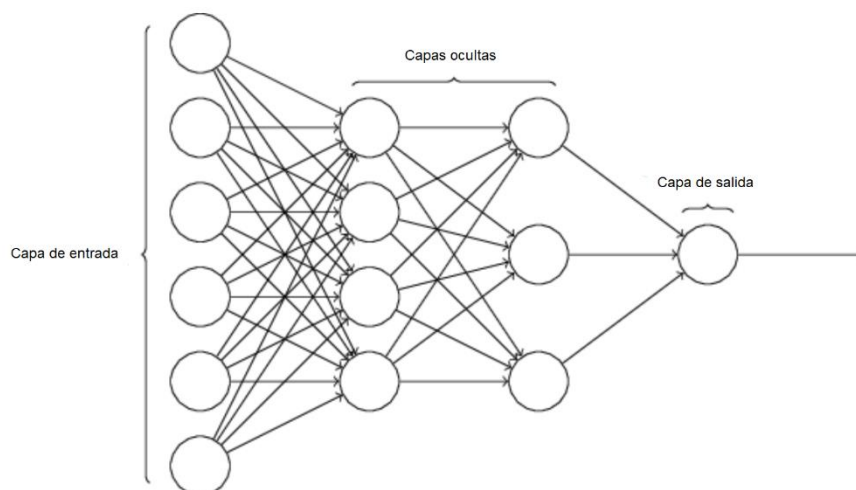


Fig. 5.1 Estructura genérica de una RNA

5.1. Origen

Con las mejoras en potencia computacional y el uso de GPUs vino un enorme avance en el desarrollo y las aplicaciones de las RNA. Del descubrimiento del perceptrón (1958) y los perceptrones multicapa (1965), a la introducción de modelos más complejos, como las redes neuronales convolucionales y redes neuronales recurrentes [10]. Actualmente las RNA, debido a estos cambios y su complejidad, están dentro del campo conocido como aprendizaje profundo.

La tecnología del aprendizaje profundo, en inglés *deep learning*, es uno de los métodos del aprendizaje automático, en inglés *machine learning*, que utiliza algoritmos para asimilar representaciones de datos. El adjetivo profundo proviene de la existencia de varias capas en estas redes y es relativamente reciente.

El avance que ha supuesto la capacidad de entrenar cientos de capas y la exploración y mejora de las redes ya existentes ha tenido un impacto directo en el sector tecnológico actual. Algunas de estas aplicaciones son: reconocimiento de voz, clasificación de imágenes, pronóstico de series temporales o diagnóstico médico.

5.2. Redes neuronales recurrentes (RNR)

Los humanos no comienzan a pensar desde cero cada segundo. A medida que se lee este proyecto, comprende cada palabra en función de su comprensión de las palabras anteriores. No se tira todo y se empieza a pensar desde cero de nuevo. Sus pensamientos tienen persistencia. Las RNA tradicionales no pueden hacer esto, y marca una gran diferencia. Una versión de las RNA, como son las RNR, abordan este problema y crean bucles que permiten que la información persista.

Estos bucles hacen que las RNR parezcan muy complejas. Sin embargo, resulta que no son tan diferentes a una RNA tradicional. Una RNR puede considerarse como copias múltiples de la misma red, cada una de las cuales pasa un mensaje a un sucesor. Observe lo que sucede si desenrollamos el bucle (ver Figura.5.2):

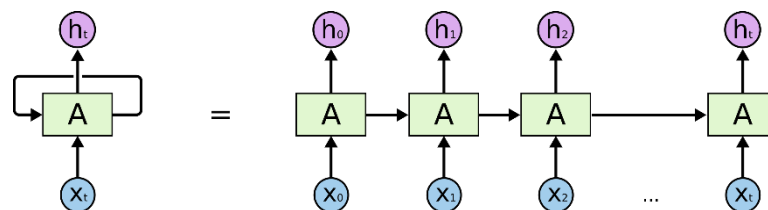


Fig. 5.2 Desglose interno de una RNA con entradas X_t , salidas h_t y red A, Fuente [5]

Esta naturaleza en cadena revela que las RNR están íntimamente relacionadas con secuencias y listas. Son la arquitectura natural de la red neuronal para usar para dichos datos. Las RNR convencionales tienen un inconveniente con las dependencias a largo plazo el cual se soluciona utilizando las redes de memoria a corto y largo plazo.

5.2.1. Redes de memoria a corto y largo plazo (LSTM)

Las redes de memoria a corto y largo plazo, en inglés *long short-term memory* o LSTM, son un tipo especial de arquitectura RNR y han logrado éxitos inmensos en el ámbito de las predicciones. Las LSTM están diseñadas explícitamente para evitar el problema de dependencia a largo plazo. Recordar información durante largos períodos de tiempo es prácticamente su comportamiento predeterminado, lo hacen de forma natural.

Como todas las RNR, las LSTM tienen la forma de una cadena de módulos o células repetitivas de red neuronal. Pero estas células en lugar de tener una sola capa de red neuronal tienen cuatro que interactúan de una manera muy especial.

El estado de una célula, que es la línea horizontal que pasa por la parte superior del diagrama, depende de tres factores o entradas en forma de vectores. Estas dependencias se pueden generalizar a cualquier problema como (ver Figura.5.3):

- El estado previo de la célula, es decir, la información que estaba presente en la memoria después del paso de tiempo anterior.
- El estado oculto anterior, es decir, la salida de la célula anterior.
- La entrada en el tiempo actual, es decir, la nueva información que se está introduciendo en ese momento.

La arquitectura de las LSTM tiene la capacidad de eliminar o agregar información al estado de la célula, todo esto regulado cuidadosamente por estructuras llamadas puertas y capas. Las puertas son capaces de realizar operaciones básicas como sumas o multiplicaciones vectoriales y las capas son conjuntos de neuronas.

Como se ha comentado anteriormente las LSTM tienen cuatro capas de redes neuronales ya aprendidas las cuales realizan las operaciones y dan sentido a esta metodología:

1. La “capa de puerta de olvido” gestiona que información se va a desechar del estado oculto anterior (f_t).

2. La “capa de puerta de entrada” decide que valores se actualizarán (i_t).
3. La tercera capa crea un vector con nuevos valores candidatos, (\tilde{C}_t), que puede ser añadido al estado.
4. Mediante la utilización de puertas los tres vectores generados anteriormente crean la ecuación de actualización de estado donde (C_t) es el nuevo estado y (C_{t-1}) el estado anterior. Por una parte se multiplica el viejo estado por (f_t) olvidando las cosas que se han decidido y se agrega el otro producto, los nuevos valores candidatos, escalados según cuánto se decide actualizar el valor de cada estado.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (5.1)$$

5. Para acabar, se decide la salida a generar de este estado (h_t). Esta salida se basa en el estado de célula actual, pero es una versión filtrada. Primero, se decide qué partes del estado de la célula generarán la salida (o_t), en la cuarta capa. Finalmente, se multiplica la selección anterior (o_t) por el estado de la célula actual (C_t), acotado entre -1 y 1 por la función \tanh , obteniendo la salida (h_t).

Los resultados exitosos tienen sentido cuando se analizan las LSTM y su complejidad. Para ayudar en la comprensión de una manera más gráfica (ver Figura.5.3).

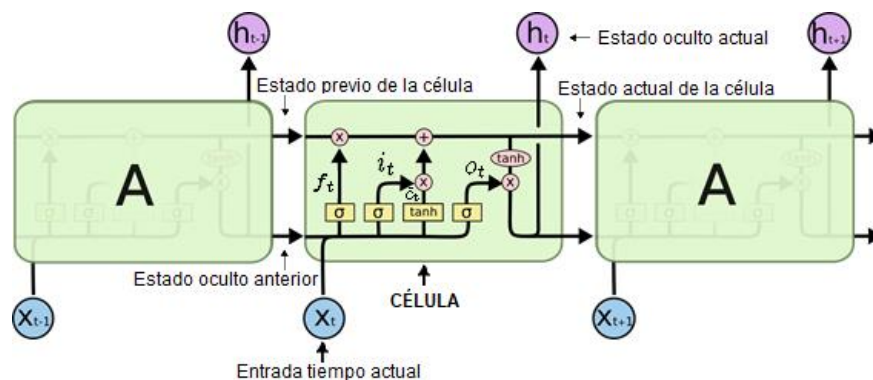


Fig. 5.3 Arquitectura celular con todas las señales vectoriales, Fuente [5]

5.3. Proceso de aprendizaje

El aprendizaje supervisado es la técnica utilizada en las LSTM para deducir una función a partir de datos de entrenamiento. Estos datos se dividen en entradas y resultados, así se consigue crear una función capaz de predecir el valor correspondiente a cualquier entrada

válida después de haber visto una serie de ejemplos, llamados datos de entrenamiento. Para ello se deben definir diversas funciones capaces de modelar esta red con precisión y reducir la diferencia entre el valor predicho y el real.

5.3.1. Función de activación

La función de activación es una ecuación matemática que, a partir de la suma ponderada de entrada y sesgos, determina la salida de una neurona. Las RNA dependen de funciones de activación no lineales, ya que la derivada de la función de activación ayuda a aprender patrones complejos en los datos a través del proceso de propagación hacia atrás [11].

Las funciones de activación utilizadas en las cuatro capas de redes neuronales que componen la célula de esta red son de dos tipos:

- La función sigmoidea (σ), utilizada en las capas uno, tres y cuatro, genera como resultado valores entre 0 y 1. Es muy útil ya que permite olvidar totalmente lo que no es importante, con un 0, y memorizar lo que sí, en diferente medida de importancia (ver Figura.5.4).
- La función tangente hiperbólica (\tanh), utilizada en la capa dos, en la salida da como resultado valores entre -1 y 1. En este caso se utiliza para evitar la acumulación de valores con órdenes de magnitud muy grandes en las operaciones realizadas por las puertas, ya que ralentiza el proceso (ver Figura.5.4).

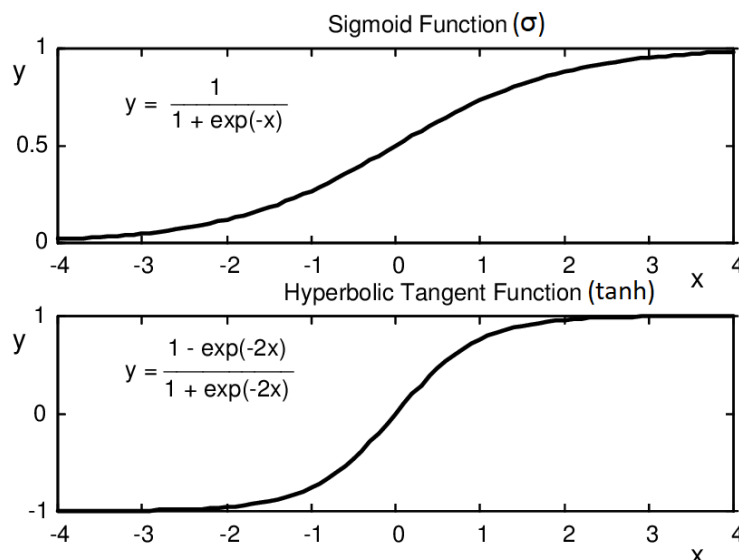


Fig. 5.4 Gráficos de las funciones Sigmoid y Tangente Hiperbólica

5.3.2. Función de pérdida

Cuando se realiza el proceso directo para generar la predicción inicial se utiliza una función de error (E), la función de pérdida, que define qué tan lejos está el resultado del valor real. La función utilizada en este proyecto es:

$$RMSE = E(\hat{y}, y) = \sqrt{\sum_{i=1}^m \frac{(\hat{y}_i - y_i)^2}{m}} \quad (5.2)$$

Donde (m) es el número total de muestras de entrenamiento, (y) es el objetivo (real) y (\hat{y}) el valor predicho. La ecuación escogida para este proyecto es la raíz del error cuadrático medio (RMSE) (Ec.5.2), utilizada para tareas de regresión.

El objetivo es encontrar un conjunto de pesos que minimice el valor de (E) en todo el conjunto de muestras de entrenamiento. La influencia de los pesos de las conexiones entre neuronas de cada capa en la función de pérdida es determinada mediante propagación hacia atrás.

5.3.3. Propagación hacia atrás

Para lograr el conjunto óptimo de pesos, se realiza la propagación hacia atrás o retro propagación. Haciendo el camino inverso, de la predicción de la red a las neuronas que generaron esa predicción. La propagación hacia atrás es un método para calcular la primera derivada de la función error, con respecto a cada peso de cada bucle, para encontrar los pesos que lleven a la función de pérdida al mínimo. Se realiza mediante el proceso matemático conocido como descenso de gradiente (GD). Cuando la red está aprendiendo, en realidad está minimizando la función error computando el valor óptimo para los parámetros de red. Como la función RMSE es una función convexa, el objetivo es encontrar el óptimo global con el GD, actualizando iterativamente los parámetros de la red.

5.3.4. Sobreajuste y sub-ajuste

Para poder comprobar que el aprendizaje es correcto, se guarda una parte de las muestras para realizar la validación, el testeo, es muy importante nunca entrenar el modelo con estas muestras ya que la validación perdería sentido. El resto se utilizarán para entrenar el modelo. Esto permitirá seguir la evolución del entrenamiento de la red para cada iteración evaluando la función error en ambos grupos y comparar su evolución.

Cuando la red neuronal es buena para aprender el conjunto de muestras de entrenamiento, pero no tiene la capacidad de generalizar ese conocimiento adquirido a ejemplos adicionales e invisibles, se dice que la red está sobre ajustada. Es reconocible por sufrir alta varianza y sesgo pequeño en la validación (ver Fig.5.5). Eso se puede evitar con el reentrenamiento de

la misma red con diferentes valores para los parámetros iniciales, con un seguimiento del error después de cada iteración del entrenamiento para detener el entrenamiento antes de que sobreajuste o también al agregar un parámetro que elimina información de manera aleatoria.

Por otra parte, el ajuste insuficiente ocurre cuando la red neuronal no puede predecir con precisión ni siquiera el conjunto de entrenamiento, sin mencionar el conjunto de validación. Esto se caracteriza por un alto error de entrenamiento y un alto error de validación (ver Fig.5.5). Se puede evitar la falta de ajuste agregando más muestras de entrenamiento [12].

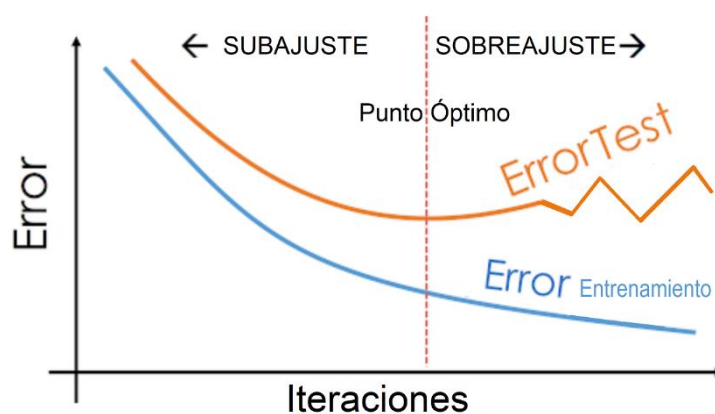


Fig. 5.5 Gráfico explicativo de los efectos de sobreajuste y sub-ajuste en aprendizaje automático

6. Análisis exploratorio de datos

Durante el análisis exploratorio de datos o EDA se tuvo el primer contacto con los datos. Se realizaron cálculos estadísticos y se crearon gráficos para descubrir tendencias, anomalías, patrones invisibles y correlaciones dentro del conjunto de datos. Un EDA proporciona información oculta sobre los datos, lo cual es de suma importancia en este caso para los pasos futuros relacionados con la construcción y el entrenamiento de la RNA.

6.1. Marco de datos

Inicialmente, el marco de datos era una tabla de *Excel* con 82 señales diferentes pertenecientes a 4 aerogeneradores. El marco de datos contiene la información desde el primer día de 2016 hasta el último día de 2017. El formato temporal era el siguiente: "AAAA-MM-DD HH:mm:ssXXX", y representa el momento preciso en el que el sistema SCADA almacenaba todas las señales en la base de datos. Las muestras de datos se recogían cada 10 minutos, siempre que el aerogenerador estuviera activo.

Lo primero que se hizo fue reestructurar los datos para que fueran más manejables y útiles. Por lo tanto, se tomaron las 417.141 muestras de las señales y se separaron en grupos, uno para cada uno de los diferentes aerogeneradores, generando cuatro marcos de datos diferentes. Esto se hizo simplemente porque cada aerogenerador tiene una ubicación y orientación diferente y generalizar todos los datos en uno puede generar incorrecciones en los resultados de la predicción.

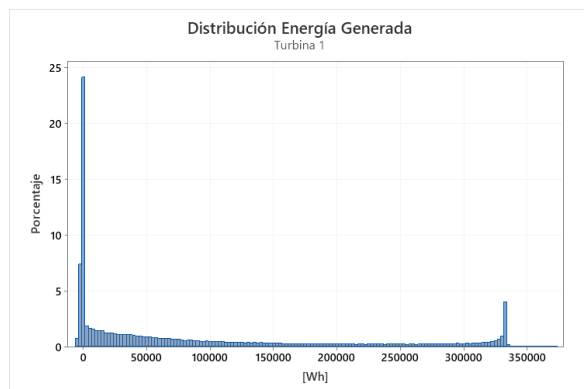
De entre todas las señales que se encontraban en el marco de datos, las seleccionadas para este proyecto fueron la energía total generada por el aerogenerador en cada intervalo de tiempo, ya que es de la que se realizarán predicciones, y la velocidad del viento media para cada intervalo de tiempo, variable de la cual depende esta producción. Se escogieron porque son señales accesibles que suelen ser comunes para todos los parques eólicos. Estas señales se encontraban en los datos como potencia activa total, en unidades energéticas de vatio-hora, y como velocidad del viento media, en metros por segundo.

6.2. Análisis objetivo

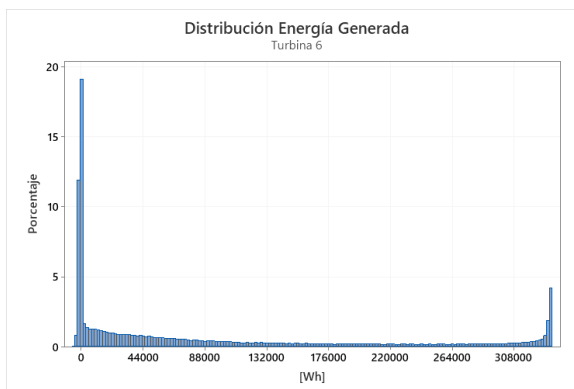
El análisis objetivo representa el primer contacto con el objeto de estudio. Da alguna información descriptiva básica sobre la distribución, variabilidad, valores atípicos, etc.

Los siguientes gráficos y la tabla son un resumen de las características básicas de los valores

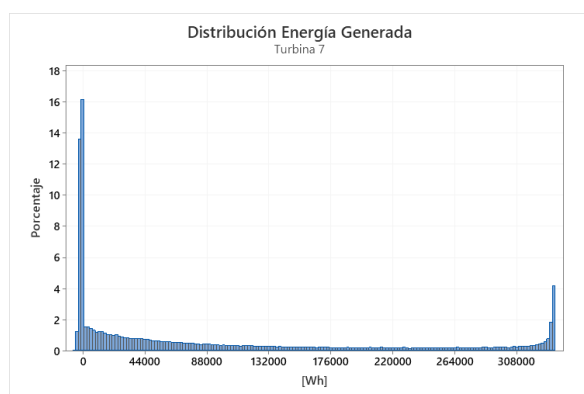
de producción energética (ver Figura.6.1 y Tabla.6.1):



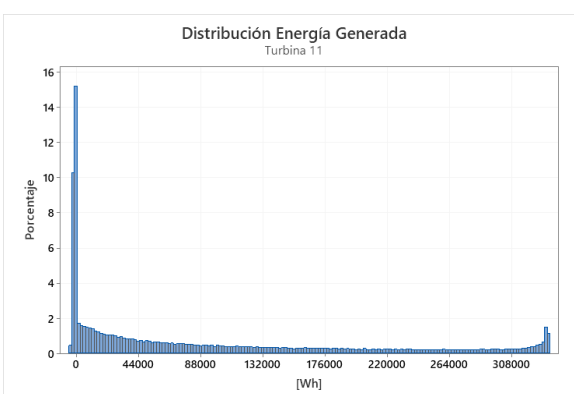
(a) Aerogenerador 1



(b) Aerogenerador 6



(c) Aerogenerador 7



(d) Aerogenerador 11

Fig. 6.1 Distribución de la energía generada separada por aerogenerador. El eje y indica el valor de densidad de probabilidad, mientras que el eje x indica los valores de consumo real.

Estadísticas

Señal	N	Media (Wh)	Desv.Est. (Wh)	Suma (Wh)	Mínimo (Wh)	Q1 (Wh)	Máximo (Wh)
Energía Generada Aerogenerador 1	104.683	86.212	109.633	9.024.961.248	-4.662	-651	371.256
Energía Generada Aerogenerador 6	102.921	89.906	113.839	9.253.166.102	-5.027	-665	334.398
Energía Generada Aerogenerador 7	104.739	91.179	114.257	9.550.025.942	-5.126	-637	334.355
Energía Generada Aerogenerador 11	104.798	90.347	105.684	9.468.149.050	-4.555	194	334.277

Tabla. 6.1 Descripción estadística de la producción energética de cada aerogenerador

La **Figura 6.1** representa las cuatro distribuciones para los distintos aerogeneradores. Se observó que eran muy similares entre ellas y como la mayor cantidad de puntos se enc en los extremos, y la mayoría de ellos en la zona de producción negativa, más del 25% de los puntos. Esta acumulación de puntos fue objeto de estudio en apartados siguientes del EDA. Se pudo intuir también la existencia de un valor atípico en la zona de máximos de la distribución del aerogenerador 1 (ver a).

La **Tabla 6.1** resume estadísticamente los datos. Se pudo observar cómo ningún aerogenerador dispone de todas las muestras para este periodo de tiempo, las cuales serían 144 diarias por un total de 2 años, que resultaría en 105.264 muestras. Esto provoca la existencia de días incompletos y se debe tener en cuenta. Se observó también que existe bastante diferencia productiva total entre los diferentes aerogeneradores según su ubicación, con una diferencia en este periodo bianual de más de 500 megavatios-hora entre el más productivo y el menos. Por último, en esta visión general, se pudo corroborar la existencia de como mínimo un valor anómalo en la turbina 1, proveniente del máximo, y que más de un 25% de los valores se encontraban entre valores negativos, observando que el límite del Q1 era negativo en la mayoría de casos, lo cual provocaba una media bastante baja para esta variable.

A continuación, mediante una tabla y diversos gráficos se observa el análisis de la señal velocidad del viento (ver Tabla.6.2 y Figura.6.2):

Estadísticas

Señal	N	Media (m/s)	Desv.Est. (m/s)	Mediana (m/s)	Mínimo (m/s)	Q1 (m/s)	Máximo (m/s)
Velocidad viento aerogenerador 1	104.683	6,0064	3,7054	5,3000	0,4000	3,1000	24,8000
Velocidad viento aerogenerador 6	102.921	5,9618	3,7482	5,2000	0,4000	3,0000	26,6000
Velocidad viento aerogenerador 7	104.739	6,1916	3,7258	5,4000	0,5000	3,3000	25,5000
Velocidad viento aerogenerador 11	104.798	5,9605	3,2298	5,4000	0,4000	3,3000	21,8000

Tabla. 6.2 Descripción estadística de la velocidad del viento de cada aerogenerador

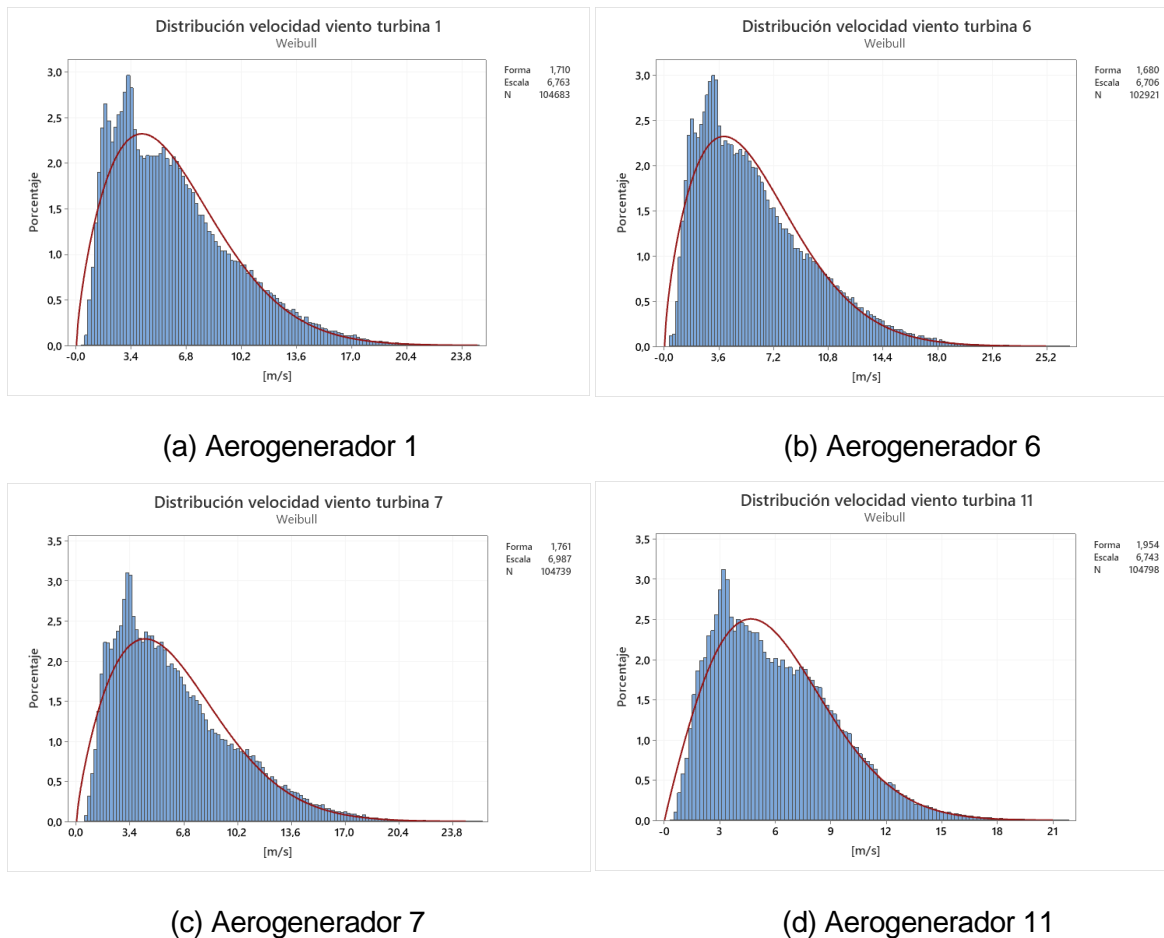


Fig. 6.2 Distribución de la velocidad del viento de cada aerogenerador. El eje y indica el valor de densidad de probabilidad, mientras que el eje x indica los valores de la velocidad.

La **Fig.6.2** grafica la distribución de las velocidades del viento y genera una distribución adaptada comúnmente utilizada para este parámetro atmosférico como es la *weibull*. Tanto los parámetros de la distribución *weibull* como los gráficos destacaban la similitud de los datos de las cuatro turbinas, a simple vista.

En la **Tabla 6.2**, que resume estadísticamente los datos de esta señal, se observaron más detalles. Primero, se observó que la mitad de los datos están por debajo de los 5 metros por segundo, eso explicó la cantidad de valores negativos en cuanto a la producción, ya que el aerogenerador empieza a producir a partir de los 4 metros por segundo. Por último, se corroboró la similitud entre los cuatro aerogeneradores ya que la media es muy similar, también es lógico ya que están separadas por unos centenares de metros entre ellas dentro del parque eólico.

6.3. Valores atípicos

La detección de valores atípicos está relacionada con el análisis objetivo, por lo tanto, información como la media o la desviación estándar, por ejemplo, encontradas durante la sección anterior fueron utilizadas en el apartado actual. En estadística, un valor atípico es un punto de datos que difiere significativamente de otras observaciones. Un punto de datos clasificado como un valor atípico puede indicar alta variabilidad en el medidor, un error experimental o incluso la desconexión del sensor.

Existen varios métodos utilizados para la detección de valores atípicos [13], algunos de ellos fueron utilizados durante el EDA.

El primer procedimiento es a través de la visualización de un diagrama de caja. Es un método para gráficamente representar grupos de datos numéricos a través de sus cuartiles. Cada "caja" está definida por la mediana, 1er y 3er cuartiles. Los diagramas de caja también pueden tener líneas que se extienden verticalmente de las cajas (bigotes) que indican la variabilidad fuera de la parte superior e inferior cuartiles. Los valores atípicos se encuentran más lejos de la mediana que los bigotes, trazados como puntos individuales.

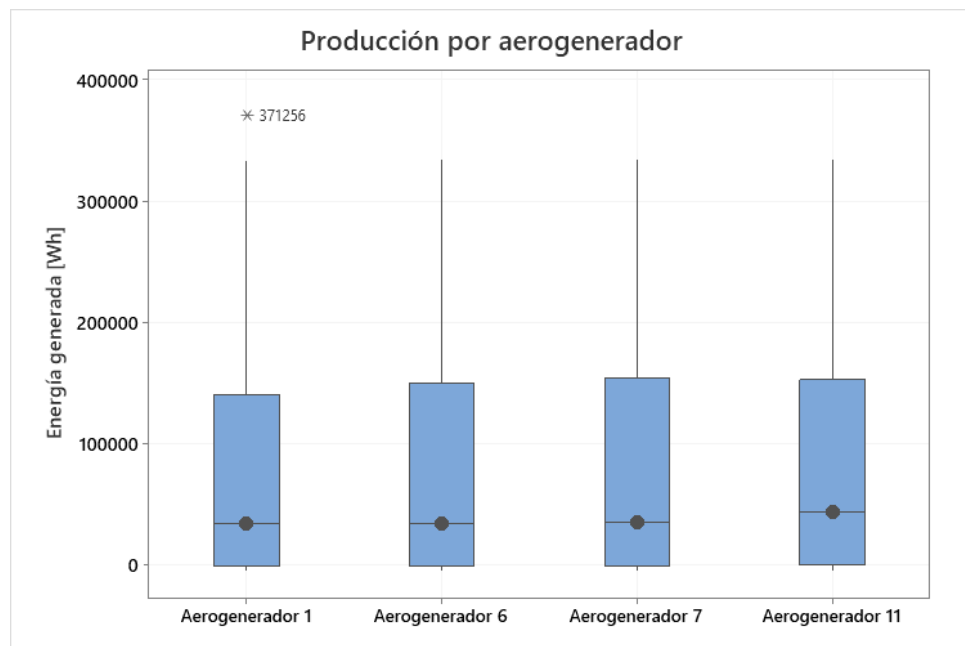


Fig. 6.3 Diagrama de cajas de la producción energética por aerogenerador

Observando la **Fig.6.3** se pudo confirmar la existencia de un único valor atípico, del que ya se sospechaba en el apartado anterior. La muestra que contenía este valor fue eliminada del marco de datos porque era físicamente imposible.

En la ficha técnica del aerogenerador, existe la curva de potencia teórica. En este proyecto, para aumentar la efectividad de las predicciones, estas se basan en datos pasados, aun así, para encontrar valores anómalos en este apartado se comparó la curva teórica con la curva real. Esta curva relaciona la velocidad del viento media de cada intervalo de 10 minutos, en metros por segundo, con la energía producida para el mismo intervalo, en vatios-hora. Para poder comparar correctamente se transformó la curva de potencia en curva de energía producida (ver Figura.6.4).

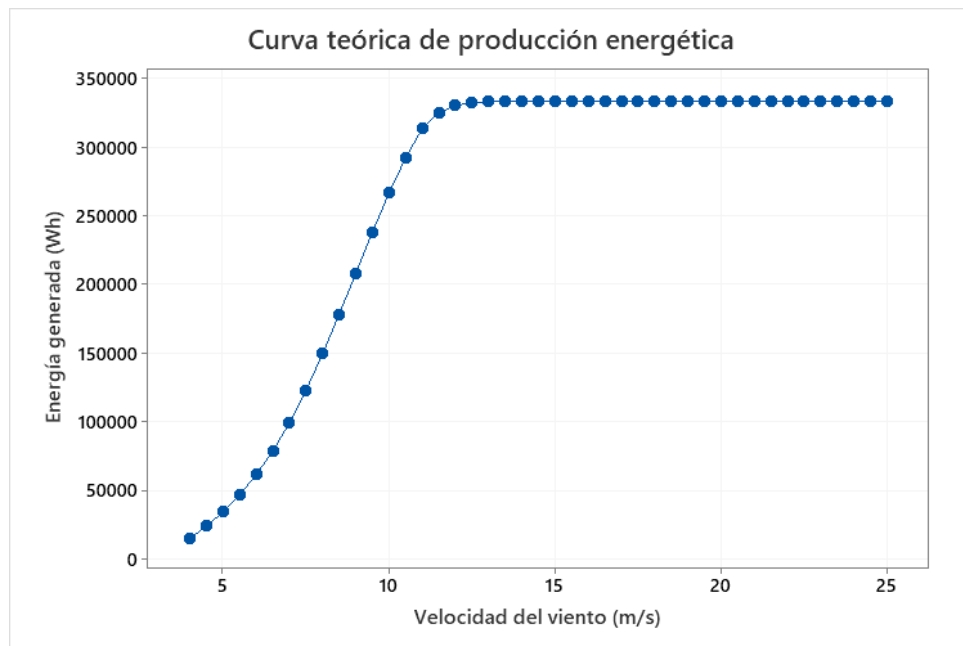
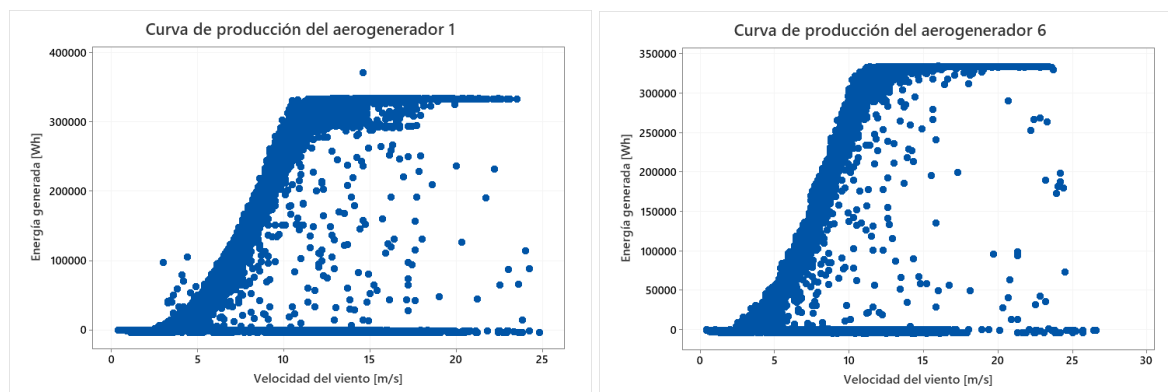


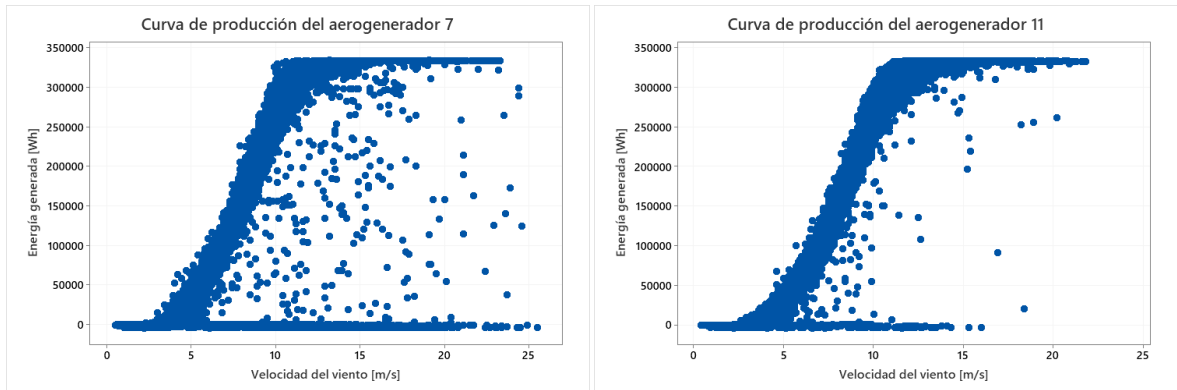
Fig. 6.4 Gráfica de curva teórica de producción del aerogenerador V90/2000

A continuación, se muestra el graficado realizado de los datos provenientes de las muestras de datos, con la misma estructura que la curva teórica representada, para cada aerogenerador (ver Figura.6.5):



(a) Aerogenerador 1

(b) Aerogenerador 6



(c) Aerogenerador 7

(d) Aerogenerador 11

Fig. 6.5 Gráficas de dispersión de la energía generada en función de la velocidad del viento para cada aerogenerador

Observando la **Figura 6.4** y la **Figura 6.5** a simple vista se apreció una similitud en la curva. En un análisis más detallado y tratando con los datos que se obtuvieron en las anteriores fases de este EDA se pudieron sacar diversas conclusiones:

- La existencia de una cantidad importante de puntos sobre la línea de energía generada nula para velocidades entre los 4 m/s y 25 m/s, velocidades mínima y máxima de producción. Se dedujo que eran valores atípicos producidos por intervalos de tiempo en los que la producción se pausó y el sistema SCADA seguía realizando medidas.
- La gran cantidad de valores negativos para diferentes velocidades. Como ya se venía viendo en los anteriores pasos.
- La existencia de valores separados de la curva de generación. Estas medidas eran físicamente posibles.
- Se observó que el aerogenerador 1 era susceptible a crear valores anómalos ubicados por encima de la curva, además bastante dispersión, como el aerogenerador 7. Sin embargo, el aerogenerador 11 tenía una recogida de datos robusta y limpia, con pocos valores fuera de la curva de producción. Además, analizando datos anteriores, el aerogenerador 11 cuenta con el mayor número de muestras de entre todos los disponibles para este periodo de tiempo.

Es importante saber que datos deben eliminarse y cuáles deben ser modificados durante el pre proceso de datos como el que se realizó. Basándose en las conclusiones anteriores se decidió realizar diferentes modificaciones en los datos para mejorar el rendimiento del futuro modelo, fueron las siguientes:

- Se eliminaron los datos correspondientes a medidas nulas o negativas de producción energética cuando la velocidad del viento era superior o igual a la mínima productiva, pero menor a la máxima admisible.
- La transformación de los datos negativos de producción energética en nulos siempre que se encontraran por debajo de la velocidad mínima productiva.

Además de las modificaciones anteriores se decidió trabajar únicamente con los datos del aerogenerador 11, que había resultado ser el más robusto a la hora de recolectar datos y eso se transferiría al modelo. Una vez realizadas las modificaciones explicadas, aplicadas sobre los datos del aerogenerador 11, el resultado aplicado sobre las gráficas anteriormente vistas fue (ver Figura.6.6 y Figura.6.7):

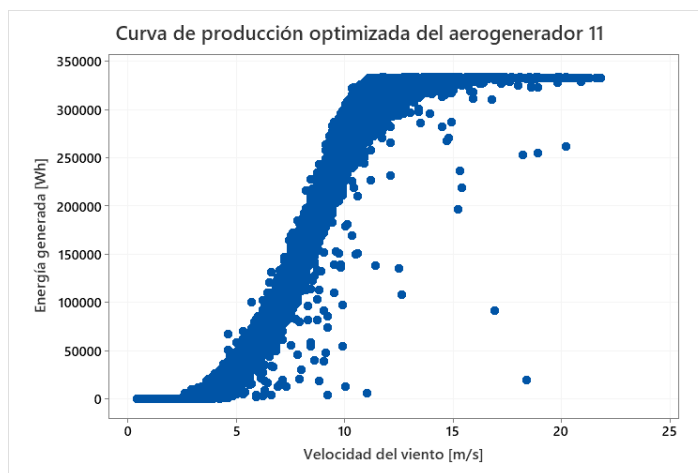


Fig. 6.6 Gráfica de la producción energética para cada medida de viento del aerogen. 11

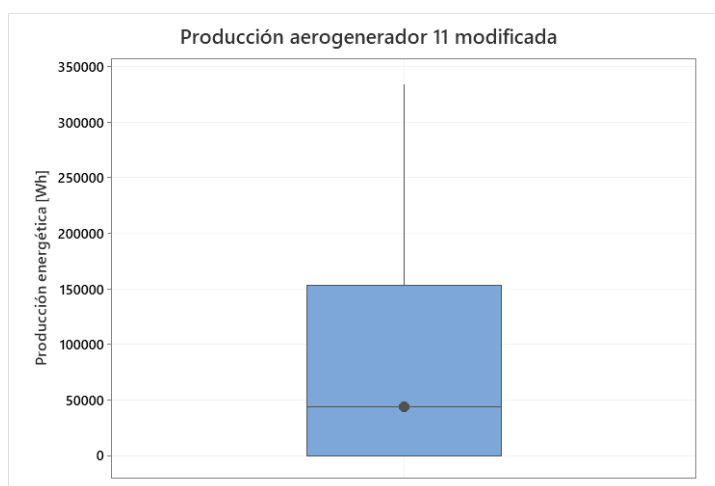


Fig. 6.7 Diagrama de caja de las muestras de producción energética del aerogen. 11

Para realizar estas modificaciones se utilizó la librería *Pandas* de *Python*. Observando las **Figuras 6.6** y **6.7** se confirmó la mejoría de los datos y la correcta aplicación de las modificaciones realizadas. Durante la eliminación de los datos erróneos se eliminaron 557 muestras de un total de 104.798, una cantidad bastante pequeña.

6.4. Eliminar datos faltantes

Es importante eliminar los datos que representan un error en la medida o están incompletos. En el apartado anterior se eliminaron muestras que no estaban tomadas correctamente por los sensores por diferentes motivos y representaban un error, en este se eliminaron los datos que resultaban incompletos.

El concepto de dato incompleto es específico para cada necesidad, en este proyecto se consideraron datos incompletos a los días que no tenían un total de 144 muestras, es decir, una muestra para cada intervalo de 10 minutos, y por lo tanto no estaba el día completo. Esta decisión evitaría falsas medidas de días incompletos con las que no se debía entrenar la RNA.

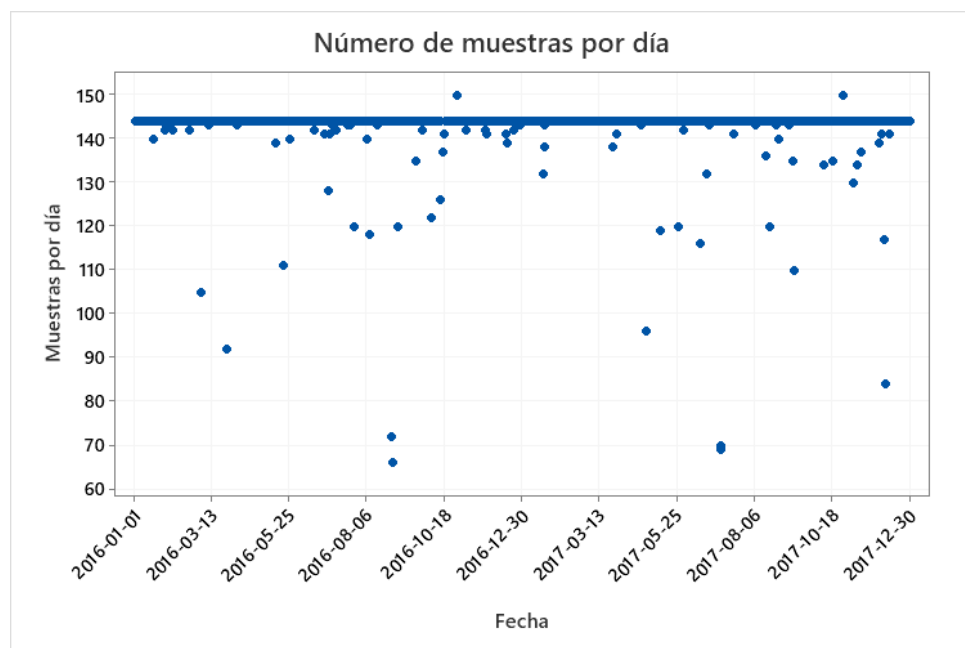


Fig. 6.8 Gráfica de serie temporal de la cantidad de muestras diarias

Se observó la distribución de estos días para obtener la distribución temporal de la cantidad de muestras por día (ver Figura.6.8). En esta se pudo ver como muchos valores se encontraban justo por debajo de la línea del día completo de muestras, en el intervalo de una hora, es decir, 6 muestras faltantes. Entonces, para evitar la pérdida desmesurada de días, se decidió poner el límite para valores con menos de 138 muestras por día. Además, se pudo

constatar la existencia de los días de cambio de hora en el equinoccio de otoño, con 150 muestras por día. Estos últimos también se conservaron ya que se encontraban dentro del margen diferencial de una hora.

La cantidad de días eliminados supuso un 4,4% del total de datos (ver Figura.6.9). Esta cantidad era perfectamente asumible y no suponía un riesgo en el fallo del entrenamiento del modelo.

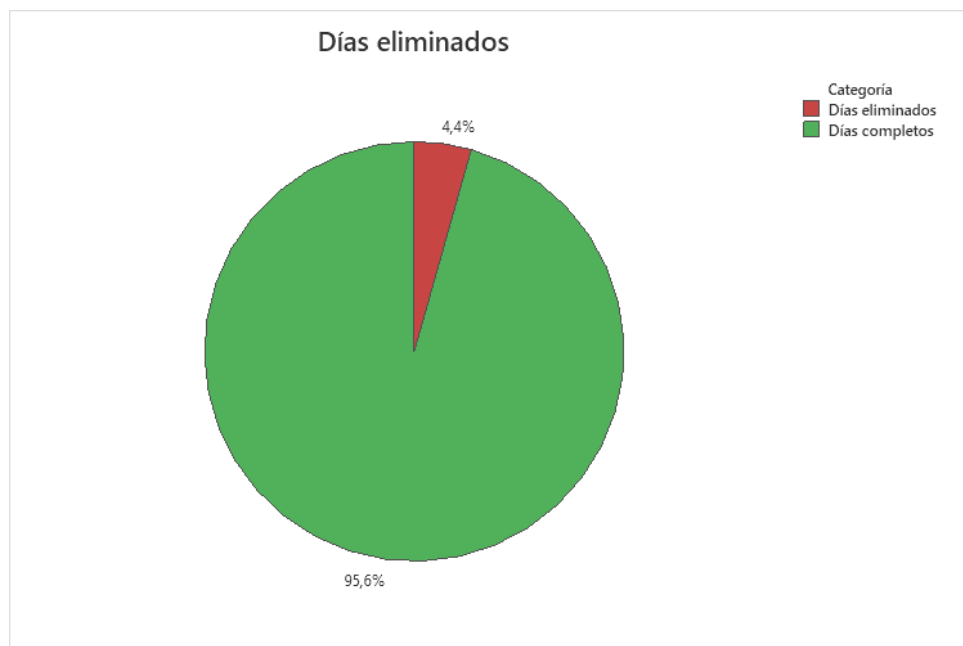


Fig. 6.8 Gráfica circular del porcentaje de días eliminados

6.5. Correlaciones

El coeficiente de correlación no es el mejor método para representar la "relevancia" de una característica, pero da una idea de las posibles relaciones dentro de los datos. La existencia de correlación entre las características del conjunto de datos y la variable objetivo puede proporcionar información sobre qué variables pueden o no ser relevantes como entrada al desarrollar un modelo para entrenar a la red. Aplicando la lógica, es obvio que tanto la producción de las muestras anteriores como la velocidad del viento de estas están relacionadas con la producción del día objetivo. En esta sección se analizó que características se debían tener en cuenta en el modelo para seleccionar correctamente las entradas de este.

Para realizar el estudio de correlación se decidió realizar cambios sobre la estructura frecuencial de los datos para reducir el error y hacer el modelo más eficiente. La existencia de muestras en intervalos de 10 minutos provocaría una gran cantidad de valores a predecir por el modelo para generar una predicción de 24 horas y, por ende, un gran error acumulado y un modelo excesivamente complejo. Estos cambios se realizaron de dos maneras distintas, que se utilizarían para la realización y comparación de dos modelos predictivos en los apartados siguientes:

1. En el modelo 1 se redujeron las muestras a nivel diario, compactando todas las muestras disponibles en grupos de 24 horas. Con respecto a la velocidad del viento, se tomó la media de las velocidades en esos periodos de tiempo. Entonces, cada muestra se componía de la energía total generada en un día y la velocidad del viento media de este mismo.
2. La segunda opción fue tomar como referencia los intervalos escogidos comúnmente para dar las predicciones de viento, es decir, intervalos de 3 horas. El concepto fue el mismo que para la primera opción, sumar todas las muestras dentro de esas 3 horas para la producción energética y realizar la media para la velocidad del viento.

Ambas modificaciones se realizaron con la función *reshape* de la librería *Pandas* de *python*. Para realizar el análisis de correlación se utilizó el coeficiente de relación de *Pearson*. Este coeficiente tiene un valor en el rango de +1 a -1, donde +1 es una relación perfecta positiva, 0 es sin relación alguna y -1 una relación totalmente negativa. En el estudio se buscaba encontrar las características previas de los datos que influyeran más sobre el valor objetivo, la producción.

6.5.1. Estudio de correlación en el modelo frecuencial 1

Este modelo temporal redujo la cantidad de datos en 144 veces aproximadamente, el número de muestras que tiene un día completo. Las previsiones creadas por el algoritmo basado en este modelo temporal serían de un único valor, que correspondería a la energía prevista total producida para el día siguiente. Al realizar el estudio de correlación se decidieron escoger las siguientes características a tener en cuenta:

- Producción energética total de las últimas 12 horas (E12hprevias).
- Producción energética total de las últimas 24 horas (E24hprevias).
- Producción energética total de hace dos días (E48hprevias).
- Producción energética total de las últimas 48 horas (Eultimas48h)
- Velocidad del viento media de las últimas 12 horas (V12hprevias)
- Velocidad del viento media de las últimas 24 horas (V24hprevias)
- Velocidad del viento media de hace dos días (V48hprevias)
- Velocidad del viento media de las últimas 48 horas (Vultimas48h)

Esta fue la tabla de resultados del estudio (ver Tabla.6.3):

Correlaciones

	Objetivo	E12hprevias	E24hprevias	E48hprevias	Eultimas48h	V12hprevias
E12hprevias	0,584					
E24hprevias	0,540	0,921				
E48hprevias	0,183	0,383	0,540			
Eultimas48h	0,412	0,743	0,878	0,877		
V12hprevias	0,579	0,981	0,908	0,396	0,743	
V24hprevias	0,534	0,903	0,985	0,553	0,876	0,914
V48hprevias	0,193	0,382	0,534	0,985	0,865	0,398
Vultimas48h	0,413	0,729	0,862	0,872	0,988	0,745
	V24hprevias	V48hprevias				
E12hprevias						
E24hprevias						
E48hprevias						
Eultimas48h						
V12hprevias						
V24hprevias						
V48hprevias	0,553					
Vultimas48h	0,881	0,881				

Tabla 6.3 Correlación del valor objetivo con características previas modelo 1

En la **Tabla 6.3** se pudo ver la relación decreciente de las características previas con el valor objetivo a medida que se alejan en el tiempo. Estas correlaciones eran especialmente débiles, pero las grandes diferencias entre ellas fueron muy útiles en cuanto a la selección de las que son más influyentes para ser utilizadas.

6.5.2. Estudio de correlación en el modelo frecuencial 2

En este modelo, más complejo que el anterior, se compactaron las muestras a grupos de 3 horas. Esto haría que el modelo tuviera que predecir 8 valores diferentes correspondientes a la energía producida en periodos de 3 horas y todos estos sumados resultarían la previsión total diaria, cada uno de estos 8 valores dependería del anterior, de aquí la mayor complejidad de este modelo. Para este modelo temporal se escogieron las siguientes características previas:

- Producción energética total de las últimas 3 horas (E3hprevias).

- Producción energética total del intervalo de las 6 horas a las 3 horas anteriores (E6hprevias).
- Producción energética total del intervalo de las 9 horas a las 6 horas anteriores (E9hprevias).
- Producción energética total de las últimas 6 horas (Eultimas6h).
- Velocidad del viento media de las últimas 3 horas (V3hprevias).
- Velocidad del viento media del intervalo de las 6 horas a las 3 horas anteriores (V6hprevias).
- Velocidad del viento media del intervalo de las 9 horas a las 6 horas anteriores (V9hprevias).
- Velocidad del viento media de las últimas 6 horas (Vultimas6h).

Esta fue la tabla de resultados del estudio de correlación (ver Tabla.6.4):

Correlaciones

	Objetivo	E3hprevias	E6hprevias	E9hprevias	Eultimas6h	V3hprevias	V6hprevias
E3hprevias	0,864						
E6hprevias	0,736	0,864					
E9hprevias	0,649	0,736	0,864				
Eultimas6h	0,829	0,965	0,965	0,829			
V3hprevias	0,846	0,967	0,851	0,729	0,942		
V6hprevias	0,719	0,846	0,967	0,851	0,939	0,870	
V9hprevias	0,634	0,719	0,846	0,967	0,810	0,729	0,870
Vultimas6h	0,809	0,938	0,940	0,817	0,973	0,967	0,967
V9hprevias							
E3hprevias							
E6hprevias							
E9hprevias							
Eultimas6h							
V3hprevias							
V6hprevias							
V9hprevias							
Vultimas6h	0,827						

Tabla 6.4 Correlación del valor objetivo con características previas modelo 2

En la **Tabla 6.4** se observó la misma tendencia que para el modelo anterior pero las relaciones entre las características y el valor objetivo, que es la energía, eran mucho más fuertes. Esto predijo un modelo más preciso a corto plazo que el anterior.

6.6. Estacionalidad

La estacionalidad se refiere a fluctuaciones periódicas. En los datos de series temporales, la estacionalidad es la presencia de variaciones que ocurren en intervalos específicos de menos de un año, como diario, semanal, mensual o trimestral. Pueden ser causadas por varios factores, como el clima, las estaciones o épocas del año. Estas suelen ser periódicas, repetitivas y patrones predecibles de una serie temporal. En este EDA se analizaron tres tipos de estacionalidad: trimestral, mensual y diaria.

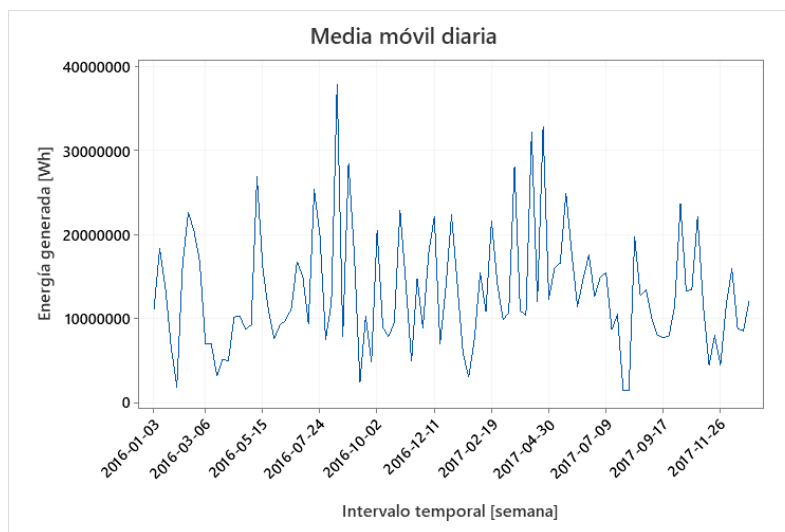


Figura 6.9 Media móvil diaria de la energía generada, en intervalos semanales

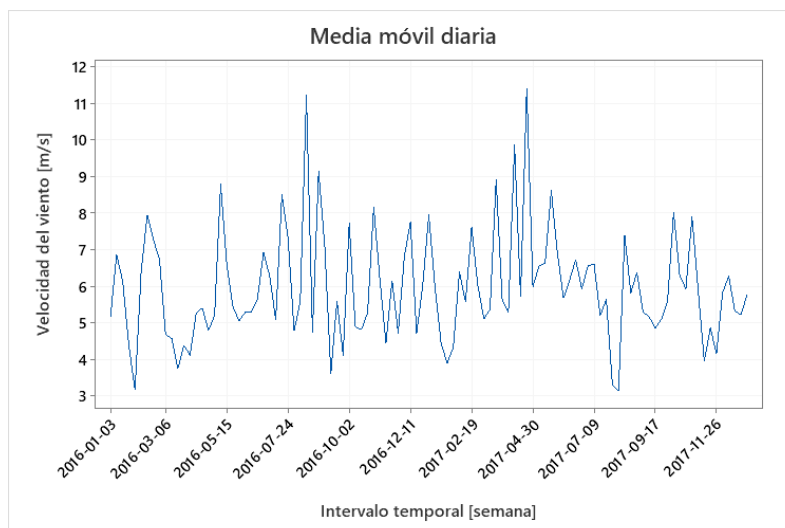


Figura 6.10 Media móvil diaria de la velocidad del viento, en intervalos semanales

En las gráficas anteriores (ver Fig.6.9 y Fig.6.10) se representó la media móvil de las dos señales con las que se ha trabajado en este proyecto, una manera de visualizar la estacionalidad de una manera general. Todos los estudios realizados en este apartado se realizaron eliminando los valores atípicos y faltantes. Se pudo observar una gran variabilidad y la relación directa de la velocidad del viento con la energía producida, pero ningún signo de estacionalidad en estas observaciones.

Hay que añadir que este apartado del EDA se vio limitado por la escasez de datos, ya que solo se disponían de dos años completos.

6.6.1. Estacionalidad trimestral

Este análisis de la estacionalidad trimestral, relacionado con las estaciones del año, tenía como función descubrir si las estaciones podían afectar a la producción energética. Para realizar el estudio siguiendo el objetivo del proyecto se utilizó la medida de la producción energética como indicador, en este caso la media diaria del mes.

Se pudo observar que en ningún caso la media de producción energética se veía afectada por la estación del año y que, de hecho, existía mucha variabilidad incluso entre mismas estaciones en los dos años (ver Figura.6.11).

Observando la misma distribución trimestral para el viento, se pudo ver el mismo comportamiento que el de los datos de producción. La variable atmosférica velocidad del viento no está afectada por las estaciones del año en esa zona (ver Figura.6.12).



Figura 6.11 Media diaria de la producción energética, en intervalos trimestrales



Figura 6.12 Media diaria de la velocidad del viento, en intervalos trimestrales

6.6.2. Estacionalidad mensual

Para realizar este estudio, el marco de datos se modificó nuevamente. Fue reconstruido mensualmente, generando la media diaria de producción energética para cada mes (ver Figura.6.13). En este segundo estudio de estacionalidad se continuó utilizando la media diaria tanto de producción como de velocidad del viento.

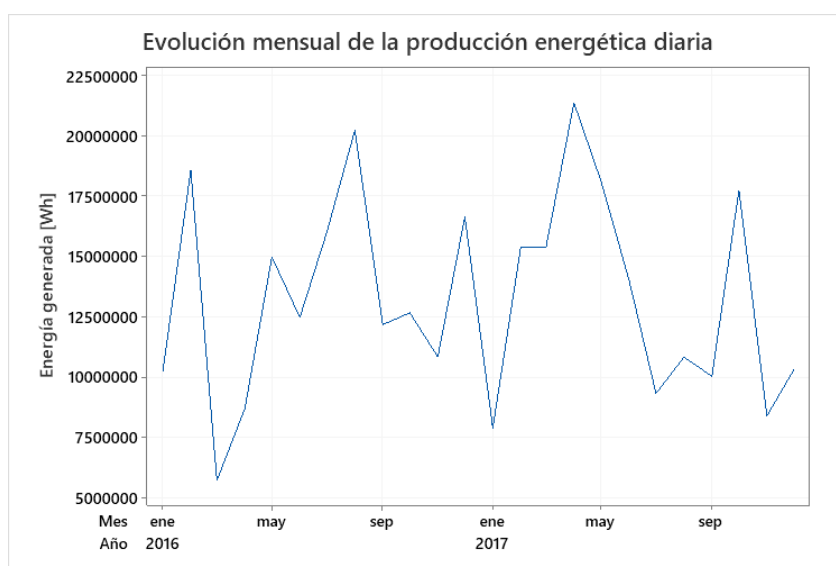


Figura 6.13 Media diaria de la producción, en intervalos mensuales

A continuación, se analizaron tanto los meses mínimos como los máximos en media de producción diaria, para ver, si se podía, su origen. En especial se analizaron los meses de marzo de 2016, enero de 2017, julio de 2017 y noviembre de 2017, correspondientes a los mínimos, y los meses de febrero 2016, agosto 2016, abril 2017 y octubre 2017, para los máximos (ver Figura.6.14 - Figura.6.17).

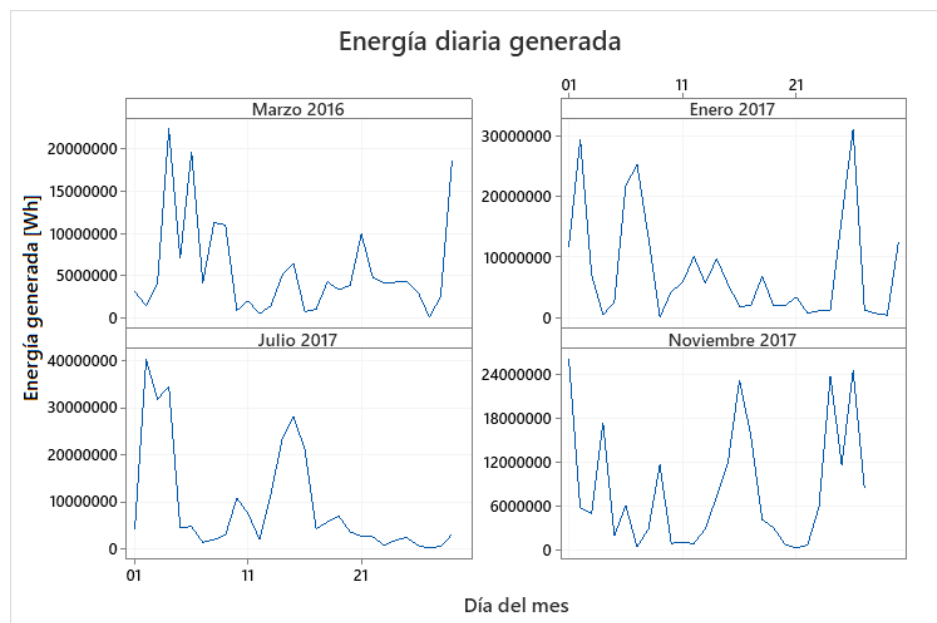


Figura 6.14 Producción diaria de los meses con media diaria mínima

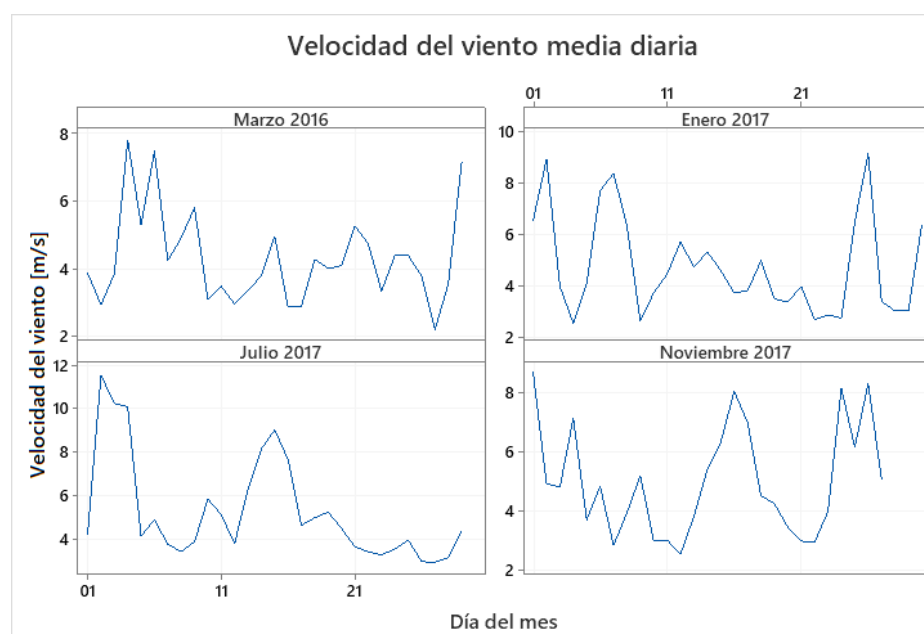


Figura 6.15 Velocidad del viento media diaria de los meses con media diaria mínima

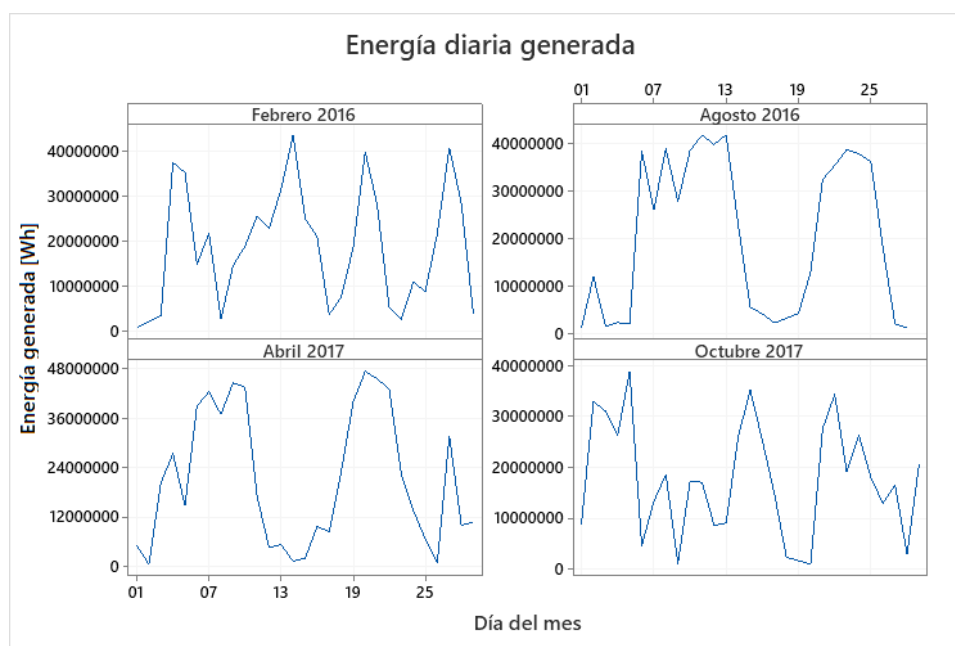


Figura 6.16 Producción diaria de los meses con media diaria máxima

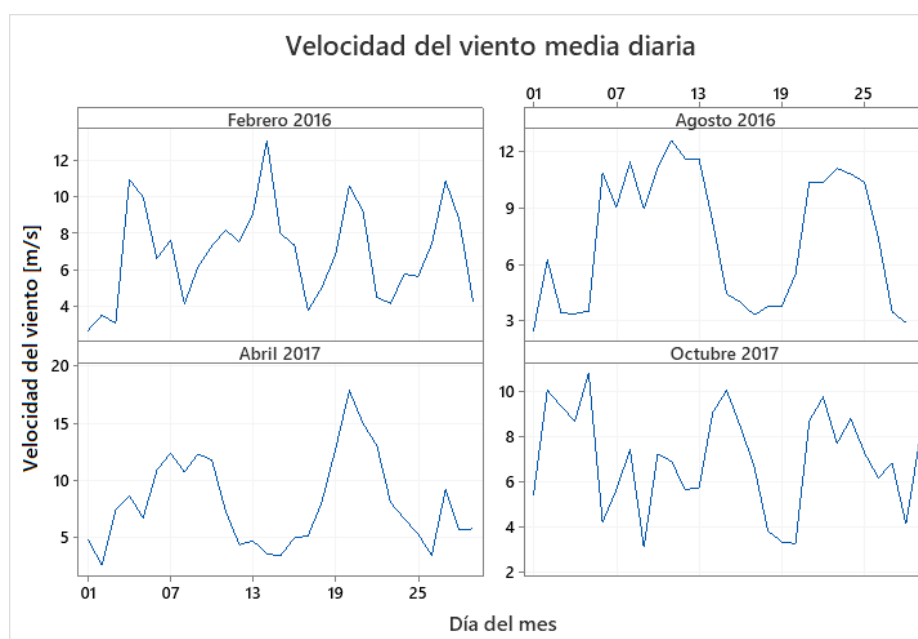


Figura 6.17 Velocidad del viento media diaria de los meses con media diaria máxima

Analizando ambos grupos se pudo observar la gran variabilidad de la producción, totalmente dependiente del viento. En la **Figura 6.15** se pudo ver como en todos los meses

representados la velocidad del viento media se encontraba en muchos días, sino la mayoría, por debajo de la velocidad mínima de producción, cosa que se reflejaba en la producción (ver Figura 6.14). Además, en la mayoría de estos meses se observó que faltaban días. Esto era producto de las modificaciones realizadas en apartados anteriores e indicó una probable desconexión del aerogenerador durante parte del día o un mal funcionamiento de este dentro de ese periodo. Por el contrario, en la **Figura 6.17** se vio como la mayoría de días del mes se encontraban por encima de la velocidad mínima de producción, y eso se reflejaba en la media de producción diaria (ver Figura 6.16).

Este segundo estudio relacionado con la búsqueda de comportamientos cíclicos tampoco mostró nada más allá que la ya conocida relación directa del viento con la producción y la alta variabilidad del viento para cualquier día y mes del año. Esto preveía una gran dificultad en el aprendizaje del modelo, pero a la vez que este modelo de red LSTM fuera el más adecuado para un caso tan complejo.

6.6.3. Estacionalidad diaria

En el último estudio de estacionalidad, se quiso ver la influencia del día y la noche sobre la velocidad del viento y la producción. Así, se modificó por última vez el marco de datos y se consideró como horario nocturno de 22 p.m. a 8 a.m. y el resto del día como horario diurno. Para comparar la producción se utilizó la media de la energía generada en periodos de 2 horas, se hizo así para evitar la desigualdad que suponía que el rango diario tuviera dos horas más que el nocturno. La comparación de la velocidad del viento se realizó simplemente con una media aritmética de todas las medidas, como en los apartados anteriores.

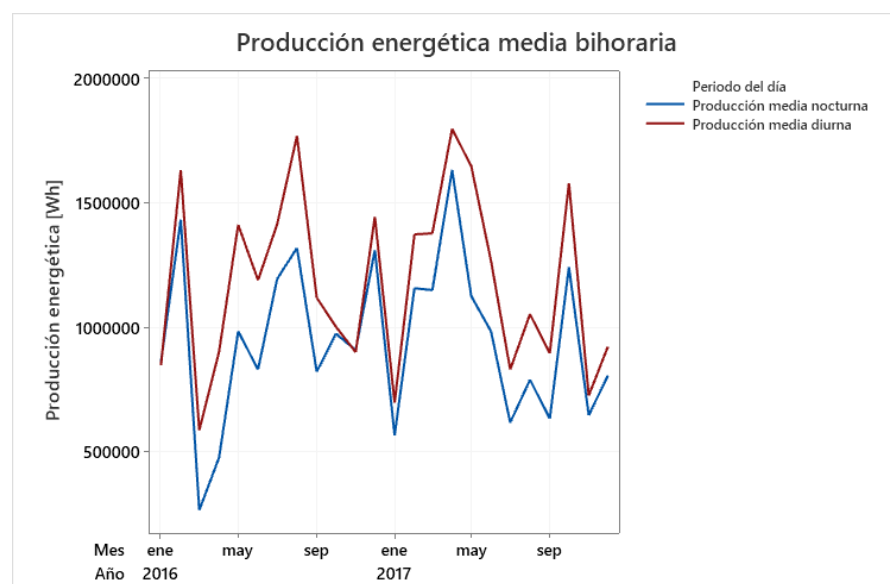


Figura 6.18 Producción media bihoraria, en intervalos mensuales, para cada horario

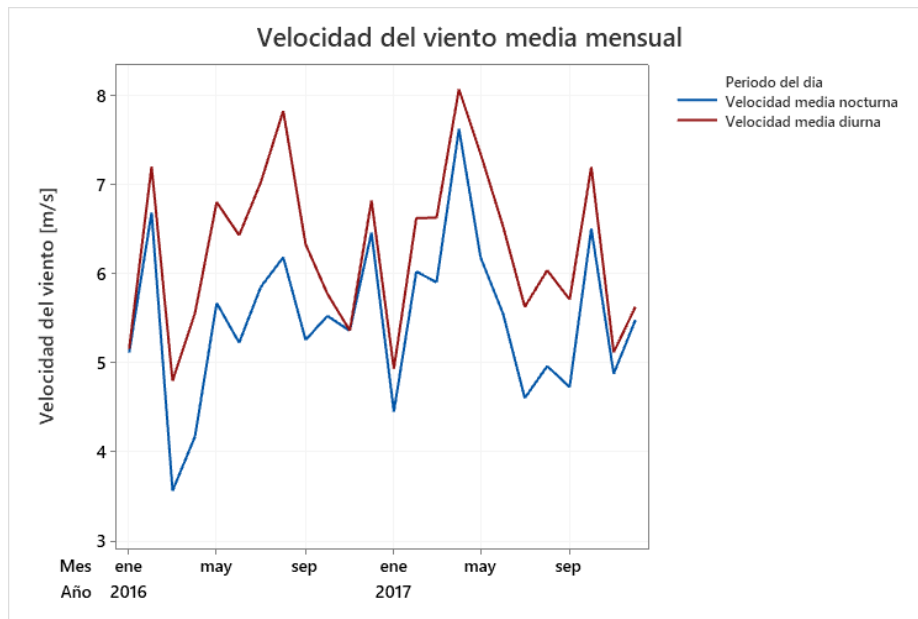


Figura 6.19 Velocidad del viento media, en intervalos mensuales, para cada horario

Finalmente, tanto en la **Figura 6.18** como en la **Figura 6.19** se pudo observar una tendencia dominante del rango diurno en detrimento del nocturno. La media de producción diurna se encuentra durante todos los meses estudiados por encima de la nocturna, al igual que la velocidad del viento. Además, se pudo apreciar una tendencia cíclica durante los meses de otoño e invierno. En estos, tanto el viento como la producción energética se emparejan y muestran diferencias mínimas, aun así, el patrón se repite en meses diferentes dentro de estas épocas.

7. Proceso de construcción

Construir una red neuronal para predecir algo en particular es una tarea no trivial. Este proceso está caracterizado por tres determinantes: interconexiones, las funciones de activación y los aspectos de aprendizaje.

La interconexión se puede definir como la forma en que los elementos de procesamiento (neuronas) de la RNA están conectadas las unas a las otras. Estas disposiciones tienen dos capas que siempre son comunes en todas las arquitecturas de red, la capa de entrada (IL) y la capa de salida (OL). La IL acepta características de entrada. Proporciona información del mundo exterior a la red, no se realiza ningún cálculo en esta capa, las neuronas aquí simplemente transmiten la información (características) a las capas ocultas. Por el contrario, la OL muestra la información aprendida por la red. El tercer tipo de capa son las capas ocultas (HL), que realizan todo tipo de cálculos sobre las características aportadas a través de la IL y transfieren el resultado a la OL. La adición de neuronas a la HL se traduce en una mayor capacidad computacional y potencia de procesamiento, sin embargo, el entrenamiento del sistema se vuelve más complejo al mismo tiempo.

El propósito de las funciones de activación, explicadas específicamente para las LSTM en el apartado **5.3.1**, es generar una salida específica de las neuronas para que no sea solo un simple problema de regresión lineal. La función está activada cuando el resultado calculado alcanza el umbral especificado [14]. La salida en esta instancia es la suma ponderada de la entrada por los pesos más el sesgo, asociado a una neurona en particular (ver Ec.7.1):

$$Salida = activación(x_1w_1 + x_2w_2 + x_3w_3 + \dots + sesgo) \quad (7.1)$$

La forma en que una RNA aprende, como se explicó anteriormente en el punto **5.3**, implica varias opciones que influyen en la velocidad y complejidad del entrenamiento, así como en la precisión en el cálculo de los valores de salida.

Los conceptos anteriores están englobados en lo que se conoce en redes neuronales como *hiperparámetros*, parámetros externos del modelo especificados previamente para ajustar la red optimizando la precisión, el tiempo de ejecución y la memoria requerida. El objetivo de esta sección era reunir la información necesaria sobre el modelado de la red y luego observar el comportamiento de los diferentes *hiperparámetros* propuestos sobre diferentes modelos, realizando un test de desarrollo. Así, se conseguiría llegar al conjunto de valores óptimos, o un rango de ellos, que ayudarían a entrenar el modelo en óptimas condiciones.

7.1. La arquitectura de la red

Las RNA generalmente están compuestas por capas de neuronas donde todas las neuronas de entrada están en una misma IL, todas las neuronas de salida están en una OL y las neuronas ocultas se distribuyen en una o más HL en el medio. Por lo tanto, el diseño de la estructura se define por: el número de capas y el número de neuronas que tiene cada capa.

La forma más confiable de configurar estos parámetros para cada problema de pronóstico específico es a través de la experimentación sistemática [15]. Sin embargo, este método puede ser complicado si se carece de experiencia, ya que hay muchas posibilidades.

Una red neuronal de una sola capa solo se puede usar para representar funciones separables linealmente, de esta manera se eligió la ya explicada red LSTM del grupo de las RNN, capaz de recordar y seleccionar información útil en grandes cantidades de datos y proyectarlo en resultados futuros. Para la implementación de la red LSTM se utilizaron las librerías *Keras* y *Tensorflow* de *Python*, las cuales permitieron la aplicación de esta red de una manera más compacta. El conjunto recurrente de células que formaban la red LSTM, explicada en profundidad en el apartado 5.2.1, fueron las HL de una RNA tradicional, lo cual permitió modificar la complejidad de esta y adecuar la IL y OL a sus requerimientos.

Se discute mucho sobre el número de neuronas necesarias para cada una de las HL, y no hay un número válido indiscutible. A pesar de la controversia, hay algunos documentos relacionados con este tema que afirman que, contra menos neuronas ocultas mejor, ya que se tiene una mayor capacidad de extraer predicciones precisas y menos sobreajuste. Para los datos observados, hay un número infinito de funciones que pasan a través de todos los pares de entrada-salida. Sin embargo, la "mejor" función no es necesariamente una que se ajuste a todos los datos observados, sino la que generaliza o predice mejor. Es común en el aprendizaje profundo seleccionar el número de neuronas en potencias de 2, el mismo formato que en las memorias utilizadas en electrónica.

Siempre es importante tener en cuenta que en el aprendizaje profundo cada problema es diferente. De este modo, se decidió ir incrementando las potencias de 2 de manera progresiva hasta que no se obtuviera mejora.

En cuanto a las otras capas, el número de neuronas de la OL depende del número de resultados que se quieren obtener y el número de neuronas de la IL de las entradas utilizadas para alimentar la red. Ambos valores fueron diferentes para cada modelo frecuencial.

7.2. Normalización de los datos

La normalización de datos se realiza antes de que comience el proceso de entrenamiento y es un aspecto muy importante. Esta hace que la red LSTM sea menos sensible a la escala de los valores de entrada. Por lo tanto, la normalización asegura un aprendizaje más rápido y que un problema de convergencia no sufra una varianza alta, lo que hace posible la optimización. La normalización es una modificación de los datos del rango original para que todos los valores estén dentro del rango de valores entre 0 y 1, donde 0 es el valor mínimo del grupo de datos y 1 el máximo. Para modificar los datos se utiliza la siguiente formula (ver Ec.7.2):

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (7.2)$$

Hay un debate que dice que es mejor tener los valores de entrada centrados alrededor del 0, este proceso es llamado estandarización, en lugar de entre 0 y 1 [16]. En este caso se pudo negar esta afirmación por un motivo de peso. Este fue relacionado con la distribución de los valores, ya que para la estandarización se recomienda que estos tengan una distribución normal y en este caso no era así, como se pudo ver en el apartado 6.2 del EDA.

Para realizar la normalización de los datos se utilizó la función *MinMaxScaler* de la librería *sklearn* de *Python*. Vale la pena mencionar que, como resultado de la normalización, la salida de la red corresponde al rango normalizado. Así, para interpretar los resultados obtenidos, esos los valores se tuvieron que volver a ajustar al rango original.

7.3. Funciones de activación

Las funciones de activación también se llaman funciones de transferencia. Determinan la relación entre entradas y salidas de una neurona y una red. Además de las ya vistas y explicadas en la teoría de las LSTM, apartado 5.3.1, entre algunas de las más utilizadas en la práctica también se incluyen:

- La función binaria o escalón es comúnmente utilizada en modelos de clasificación. El resultado será siempre un 0 o un 1, sin más opciones (ver Figura.7.1). Estas características hicieron que no fuera necesario ni siquiera plantearse su uso en este modelo regresivo, por lo tanto, no se entrará en más detalles sobre ella.

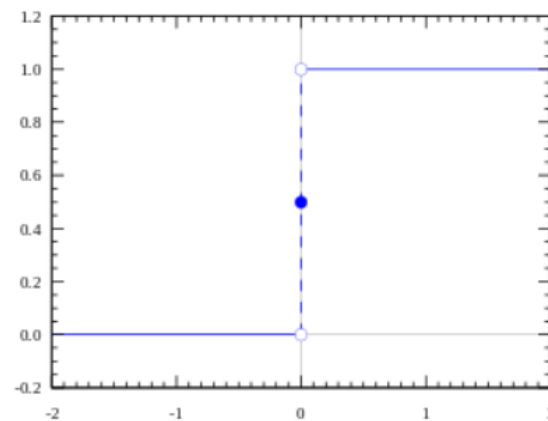


Figura 7.1 Función binaria

- La función lineal, cuya salida siempre es proporcional a la entrada y permite diferentes valores, lo cual la diferencia de la función binaria (ver Figura 7.2). Sin embargo, esta función no permite la retro propagación, ya que, al calcular la derivada en el método de GD, resulta una constante y no permite progresar. Además, utilizar esta función es como eliminar todas las capas de la red y transformarlas en una única, ya que el resultado siempre será proporcional a la entrada.

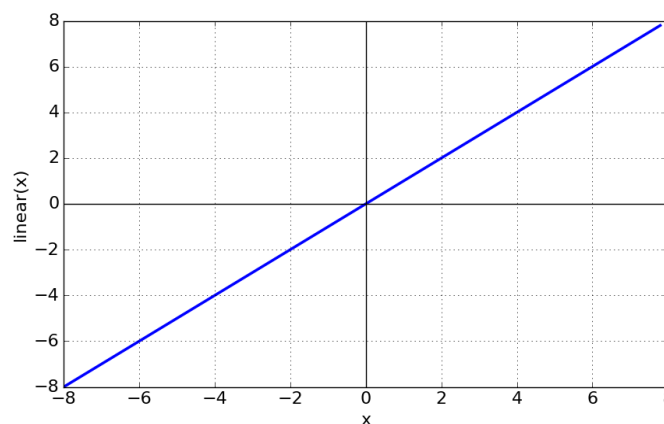


Figura 7.2 Función lineal

- La función unidad lineal rectificada (ReLU) es similar a la función lineal, pero sí que permite la retro propagación, aunque su forma no lo aparente (ver Figura 7.3). Además de ser una función no lineal esta ayuda al modelo a converger rápidamente. El único inconveniente es que si se trabajan con negativos no se ejecuta la propagación hacia atrás de manera correcta, este problema es llamado la *dying ReLU*.

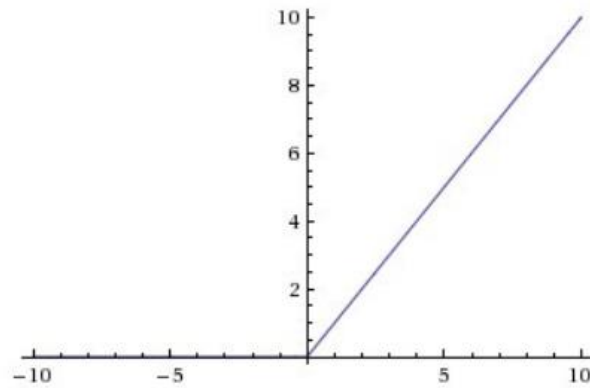


Figura 7.3 Función ReLu

Una red puede tener diferentes funciones de activación para diferentes neuronas en la misma capa. Sin embargo, en este caso, se utilizó la misma función para toda la capa, debido a la estructura de la librería que se utilizó para compilar esta red. Teniendo en cuenta las funciones de activación estudiadas en este apartado, en el 5.3.1, y sus características, se pudo decidir con exactitud cuáles eran más convenientes para el modelo.

La biblioteca utilizada para generar la red LSTM permitía aplicar algunos cambios sobre la estructura vista en la teoría de la sección 5.2.1, sin modificar su esencia. El cambio que se realizó fue la sustitución de la función por la cual se multiplica la salida de la capa 4, substituyendo la original tanh por una ReLu. Esto permitía conservar la distribución no normal de los datos comentada anteriormente, aumentando la precisión y reduciendo el tiempo de cálculo. Además, con las modificaciones realizadas sobre las muestras con valores negativos en el apartado 6.3 del EDA, se evitó el problema *dying ReLu*.

Para la capa de salida, se decidió comparar el rendimiento de las funciones ReLu y sigmoid en el test de desarrollo.

7.4. Algoritmos de optimización del aprendizaje

El entrenamiento de la red neuronal es un problema de minimización no lineal en el que los pesos de la red se modifican iterativamente para minimizar, en este caso, la función error entre los valores de salida predichos y los reales. Los algoritmos de optimización son los encargados de realizar la retro propagación mediante el proceso matemático de GD, ya explicado en el apartado 5.3.3. Ninguno de ellos garantiza una solución óptima para un problema general, por lo tanto, se eligieron los más adecuados.

Entre los algoritmos de optimización que aplican el GD se encuentra el descenso de gradiente estocástico (SGD). Este se caracteriza por la actualización de las variables del modelo para cada lote de datos, por esto es conocido como el algoritmo de optimización en línea [6]. Para este algoritmo, una ratio de aprendizaje se debe especificar, el cual determina la magnitud del cambio sufrido por los pesos en cada iteración. El entrenamiento es bastante sensible a este *hiperparámetro* ya que las tasas de aprendizaje más pequeñas tienden a ralentizar el proceso de aprendizaje, mientras que las más grandes pueden causar oscilaciones que no permitan acercarse al óptimo. Una posible forma de mejorar el aprendizaje es incluir dos términos adicionales: el momento, que provoca que los próximos valores de los pesos cambien en más o menos la misma dirección que los anteriores, y así reducir el efecto de oscilación de mayores tasas de aprendizaje; y la decadencia, que es lo que hace que la tasa de aprendizaje disminuya por un factor en cada iteración, aumentando la precisión al final mientras se aproxima al valor óptimo.

Nuevamente, los mejores valores para los parámetros de aprendizaje generalmente se determinan a través de la experimentación, entre un grupo seleccionado de valores. Entre los valores seleccionados se encontraron: 0.05, 0.01 y 0.001, para la tasa de aprendizaje; 0.9, para el momento y 1e-6, para la decadencia. Estos valores se escogieron basados en experiencias previas observadas, para que la experimentación no fuera tan costosa [6].

El desafío de variar las ratios de aprendizaje durante el entrenamiento de la red es que sus *hiperparámetros* deben ser definidos con anterioridad y dependen en gran medida del tipo de modelo y problema para su efectividad. Por el contrario, los algoritmos optimizadores de GD adaptativos como *Adagrad*, *Adadelta*, *RMSprop* y *Adam*, proporcionan una alternativa al SGD clásico. Estos métodos de ratio de aprendizaje por parámetro brindan un enfoque heurístico sin requerir un trabajo de ajuste manual de *hiperparámetros* para la ratio de aprendizaje variable [6].

Para disponer de todas las opciones posibles y no descartar sin haber probado todo, se decidió incluir en el test de desarrollo todos los algoritmos optimizadores adaptativos mencionados anteriormente. Además, como ya se explicó en la sección 5.3.2, la función error que se usó para optimizar el modelo fue la raíz del error cuadrático medio. Como métrica comparativa, se decidió utilizar tanto el error relativo medio (ERM) como el error medio absoluto (MAE).

7.5. Número de épocas y tamaño del lote

Los algoritmos de optimización del aprendizaje basados en el GD actualizan las variables del modelo a la vez que realizan el entrenamiento, los *hiperparámetros* que dicen cada cuanto pasa esto se explican a continuación.

Primero hay que entender el significado de muestra. El marco de datos que se estudia está compuesto por filas de datos correspondientes al mismo instante, llamadas muestras. El tamaño del lote es un *hiperparámetro* que define el número de muestras a recorrer antes de actualizar los parámetros internos del modelo. Por otro lado, el número de épocas es un *hiperparámetro* que define el número de veces que el algoritmo optimizador del aprendizaje recorrerá todo el conjunto de muestras de entrenamiento. Una época se puede componer de uno o más lotes [7].

La forma en que se combinan estos *hiperparámetros* es imaginable como un bucle que recorre el número de épocas en donde cada ciclo recorre todo el conjunto de datos de entrenamiento. Dentro de este bucle hay otro que itera sobre cada lote de muestras, donde uno lote tiene el número específico de "tamaño de lote" de muestras. Al final de cada lote, las predicciones se comparan con las variables de salida esperadas y se calcula un error. A partir de este error, el algoritmo de aprendizaje se usa para mejorar el modelo, en este caso el GD.

Cuando el lote es del tamaño de una muestra, el algoritmo de aprendizaje se denomina de aprendizaje en línea. En cambio, si el tamaño del lote es más de una muestra y menor que el tamaño del conjunto de datos de entrenamiento, el algoritmo de aprendizaje se denomina descenso de gradiente de mini lote. El tercer y último caso, donde la medida del lote es la misma que la longitud de la muestra de datos, se descartó, ya que la cantidad de muestras de los datos disponibles era demasiado grande para utilizar este método. En el caso del descenso de gradiente de mini lotes, los tamaños de lotes populares incluyen 16, 32 y 64 muestras. Estos fueron analizados en el test de desarrollo, junto a la opción de tamaño del lote de una única muestra.

El número de épocas es tradicionalmente grande, a menudo cientos o miles, lo que permite al algoritmo de aprendizaje ejecutarse hasta que el error del modelo se haya minimizado lo suficiente. Este se establece comúnmente en 10, 100, 500, 1.000, e incluso más grande. Para el caso de estudio, los siguientes valores fueron seleccionados para ser analizados durante la prueba de desarrollo: 10, 25, 50, 100 y 150.

7.6. Creación de sub-modelos

En una parte anterior del proyecto, sección 6, se realizó un análisis exploratorio de los datos,

con el objetivo de descubrir cualquier tipo de información relevante sobre el conjunto, como las correlaciones, tendencias, etc. Es durante la creación de los diferentes modelos donde esta información es de suma importancia.

Dentro de la sección mencionada anteriormente, se decidió agrupar el marco de datos y transformar la frecuencia de estos en dos nuevos modelos frecuenciales. Bajo estas dos divisiones frecuenciales se crearon los modelos que formarían parte del estudio posterior. En ambos casos se decidió crear modelos compactos, con pocas entradas y que estas fueran relevantes.

7.6.1. Variaciones basadas en el modelo frecuencial 1

Primero se debe recordar que la frecuencia de las muestras en este modelo es diaria. Teniendo en cuenta el EDA realizado se decidió crear los siguientes modelos con sus respectivas características de entrada.

Modelo 1a

- Producción energética total de las últimas 12 horas (E12hprevias).
- Producción energética total de las últimas 24 horas (E24hprevias).
- Velocidad del viento media de las últimas 12 horas (V12hprevias).
- Velocidad del viento media de las últimas 24 horas (V24hprevias).

Modelo 1b

- Producción energética total de las últimas 12 horas (E12hprevias).
- Velocidad del viento media de las últimas 12 horas (V12hprevias)

Modelo 1c

- Producción energética total de las últimas 24 horas (E24hprevias).
- Velocidad del viento media de las últimas 24 horas (V24hprevias).

Modelo 1d

- Producción energética total de las últimas 48 horas (Eultimas48h).
- Velocidad del viento media de las últimas 48 horas (Vultimas48h).

La salida de todos estos modelos es única, es decir, un único valor que correspondería a la producción total energética del día siguiente.

7.6.2. Variaciones basadas en el modelo frecuencial 2

En este modelo frecuencial se tomó como referencia la frecuencia común para las predicciones del viento, en intervalos de 3 horas. Según la información adquirida en el EDA, los modelos creados fueron los siguientes, con sus características de entrada.

Modelo 2a

- Producción energética total de las últimas 3 horas (E3hprevias).
- Producción energética total del intervalo de las 6 horas a las 3 horas anteriores (E6hprevias).
- Velocidad del viento media de las últimas 3 horas (V3hprevias).
- Velocidad del viento media del intervalo de las 6 horas a las 3 horas anteriores (V6hprevias).

Modelo 2b

- Producción energética total de las últimas 3 horas (E3hprevias).
- Velocidad del viento media de las últimas 3 horas (V3hprevias).

Modelo 2c

- Producción energética total de las últimas 6 horas (Eultimas6h).
- Velocidad del viento media de las últimas 6 horas (Vultimas6h).

Modelo 2d

- Producción energética total de las últimas 3 horas (E3hprevias).
- Producción energética total de las últimas 6 horas (Eultimas6h).
- Velocidad del viento media de las últimas 3 horas (V3hprevias).
- Velocidad del viento media de las últimas 6 horas (Vultimas6h).

En estos casos, como ya se comentó en el EDA, la salida era más especial. El modelo generaba 8 salidas correspondientes a los 8 intervalos de 3 horas que tiene un día, uno dependiente del anterior.

7.6.3. Transformación en aprendizaje supervisado

El aprendizaje supervisado, como ya se ha mencionado anteriormente, es la metodología utilizada para el entrenamiento de la red LSTM. Esta metodología tiene su base en aprender de experiencias reales previas. Entonces para ello se tenían que transformar los datos de manera que se aprovecharan estos al máximo.

Sabiendo cuales eran las estructuras de los datos de entrada de los modelos, se utilizó la misma función *reshape* que para el estudio de correlación para modelar todas las entradas y salidas. Esto supuso la creación de dos modelos supervisados, uno para cada modelo frecuencial.

En el caso del modelo diario, la salida correspondía a la energía producida el día siguiente, que se utilizaría para entrenar la red como el resultado a obtener. Por otro lado, se añadieron las entradas, que correspondían a todas las características de energía generada y velocidad del viento media de los días o periodos previos al día de la salida. Así, el resultado fue 699 muestras, una para cada día, con la siguiente estructura (ver Figura.7.4):

Fecha	Energía	E12hprevias	E24hprevias	Eultimas48h	V12hprevias	V24hprevias	Vultimas48h
2016-01-01 00:00:00+00:00	9821006	nan	nan	nan	nan	nan	nan
2016-01-02 00:00:00+00:00	6770617	7.72133e+06	9.82101e+06	nan	6.91806	5.66667	nan
2016-01-03 00:00:00+00:00	16959362	12450	6.77062e+06	1.65916e+07	2.16806	3.94028	4.80347
2016-01-04 00:00:00+00:00	39174390	1.67141e+07	1.69594e+07	2.373e+07	9.32639	5.90208	4.92118
2016-01-05 00:00:00+00:00	29104329	2.00933e+07	3.91744e+07	5.61338e+07	10.3722	10.2153	8.05868

Figura 7.4 Primeras 5 muestras del marco de datos del aprendizaje supervisado para el modelo frecuencial 1. Las energías se encuentran en Wh y las velocidades en m/s.
Fuente: *Python*

Para el modelo frecuencial en intervalos de 3 horas, la salida correspondía a la energía generada en las 3 horas siguientes al momento de la predicción y los 7 intervalos siguientes hasta conformar las 24 horas. Las entradas de este correspondían a características previas de energía generada y velocidad del viento media, como para el caso anterior. El resultado en este caso fueron 5.782 muestras, correspondientes a todos los periodos de 3h, con la siguiente estructura (ver Figura.7.5):

Fecha	Energía+24h	Energía+21h	Energía+18h	Energía+15h	Energía+12h	Energía+9h	Energía+6h	Energía	E3hprevias	E6hprevias	Eultimas6h	V3hprevias	V6hprevias	Vultimas6h
2016-01-01 00:00:00+00:00	1.47395e+06	1.60258e+06	2.4506e+06	2.1942e+06	771428	703389	77889	546970	nan	nan	nan	nan	nan	nan
2016-01-01 03:00:00+00:00	2.52263e+06	1.47395e+06	1.60258e+06	2.4506e+06	2.1942e+06	771428	703389	77889	546970	nan	nan	4.62778	nan	nan
2016-01-01 06:00:00+00:00	3.5213e+06	2.52263e+06	1.47395e+06	1.60258e+06	2.4506e+06	2.1942e+06	771428	703389	77889	546970	624859	3.00556	4.62778	3.81667
2016-01-01 09:00:00+00:00	533662	3.5213e+06	2.52263e+06	1.47395e+06	1.60258e+06	2.4506e+06	2.1942e+06	771428	703389	77889	781278	4.98333	3.00556	3.99444
2016-01-01 12:00:00+00:00	180583	533662	3.5213e+06	2.52263e+06	1.47395e+06	1.60258e+06	2.4506e+06	21941	771428	703389	1.47482e+06	5.04444	4.98333	5.01389

Figura 7.5 Primeras 5 muestras del marco de datos del aprendizaje supervisado para el modelo frecuencial 2. Las energías se encuentran en Wh y las velocidades en m/s.
Fuente: *Python*

En las **Figuras 7.4 y 7.5** se puede observar como faltan valores en las dos primeras filas. Esto sucedió al calcular características sobre experiencias previas, cuando estas no existían, es decir, no se disponían de esos datos. Por lo tanto, esas dos primeras filas fueron eliminadas en el momento del estudio. Los códigos utilizados para realizar ambos marcos de datos se encuentran explícitamente en el *Anexo A*.

7.7. Test de desarrollo

El test de desarrollo es un proceso en el que se selecciona una porción de muestras representativa del conjunto y se entrena a la red con estas para realizar la búsqueda de *hiperparámetros*, así como la selección óptima de las entradas y la frecuencia de datos. Las ventajas de usar solo una porción de las muestras del conjunto de datos, es que el tiempo computacional se reduce significativamente y se puede usar más tiempo probando una mayor diversidad de modelos y combinaciones de *hiperparámetros* para la red. En este proyecto, como el número de muestras de las que se disponía no era tan grande, se decidió ejecutarlo con la cantidad total de estas. Además, como ya se ha comentado anteriormente, se dejó lugar a optimizar el modelo más allá de los valores de los *hiperparámetros* acotados anteriormente, en caso de que se viera una potencial mejoría.

7.7.1. Indicador clave de rendimiento (KPI)

Cuando se piensa en cómo evaluar el desempeño de un RNA hay varios aspectos a estudiar. En consecuencia, se decidió aplicar la idea de crear un indicador clave de rendimiento (KPI), un coeficiente para comparar de manera justa los diferentes modelos. Este KPI, se compuso de dos variables; precisión y tiempo computacional. El objetivo era encontrar el conjunto de *hiperparámetros* que minimizaran este KPI.

Obviamente, la precisión con la que la red es capaz de aproximar la salida a los valores objetivo reales, es el aspecto más importante. A menudo, se define una medida de precisión en términos del error de pronóstico, llamada métrica. Existen varias métricas, cada una con algunas ventajas y limitaciones. Para este problema en particular, como se mencionó anteriormente, se seleccionaron dos: ERM y MAE (ver Ecuaciones 7.3 y 7.4). Donde y es el valor objetivo, \hat{y} es el valor predicho y N representa el número total de muestras con las que se entrena la red. Para utilizar el MAE de una manera más óptima, se decidió cambiar sus unidades de Wh a MWh, que es la unidad con la que se calcula el precio de la energía.

$$ERM = \frac{1}{N} \sum_{i=1}^N \frac{\|y_i - \hat{y}_i\|}{y_i} \quad MAE = \frac{1}{N} \sum_{i=1}^N \|y_i - \hat{y}_i\| \quad (7.3) \text{ y } (7.4)$$

El tiempo de computación, indica indirectamente la complejidad del modelo, a más complejidad, más cálculos y más tiempo. Para comparar, se decidió escoger como indicador temporal el tiempo medio necesario para realizar una época (TME), en segundos.

La fórmula del indicador es la siguiente (ver Ec.7.5):

$$KPI = MAE * 0,3 + ERM * 0.6 + TME * 0.1 \quad (7.5)$$

7.7.2. Pasos

Para afrontar un análisis tan grande de una manera óptima, se decidió encontrar en primer lugar los *hiperparámetros* que tuvieran gran influencia sobre todos los modelos, ya fuera positiva o negativa, para intentar reducir la medida del análisis. De esta manera se intentó conseguir llegar al punto óptimo eliminando *hiperparámetros* candidatos en cada paso.

7.7.2.1. Paso 1: Algoritmo optimizador y funciones de activación

El test de desarrollo se empezó analizando uno de los puntos clave, el algoritmo optimizador del aprendizaje y su relación con las funciones de activación. Ambos, determinan el formato de la salida, de aquí su importancia. Este primer análisis se realizó por separado para ambas estructuras frecuenciales.

Modelo frecuencial 1

Para poder valorar todas las posibilidades, se realizaron 56 pruebas. Posteriormente, se analizó el valor y la distribución del KPI para cada función de activación y algoritmo optimizador del aprendizaje.

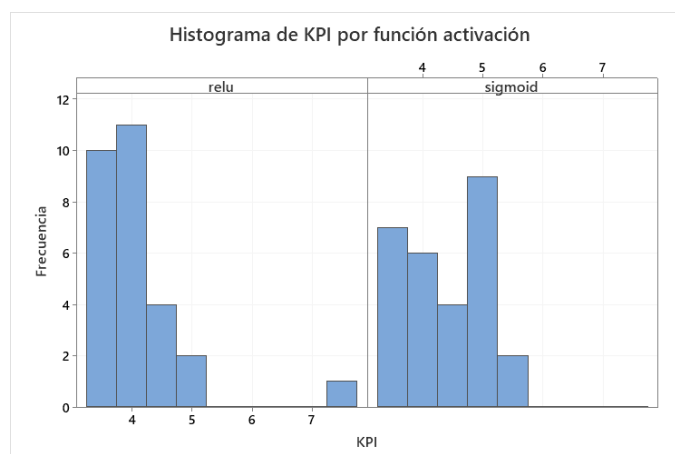


Figura 7.6 Distribución frecuencial de KPI para ambas funciones de activación

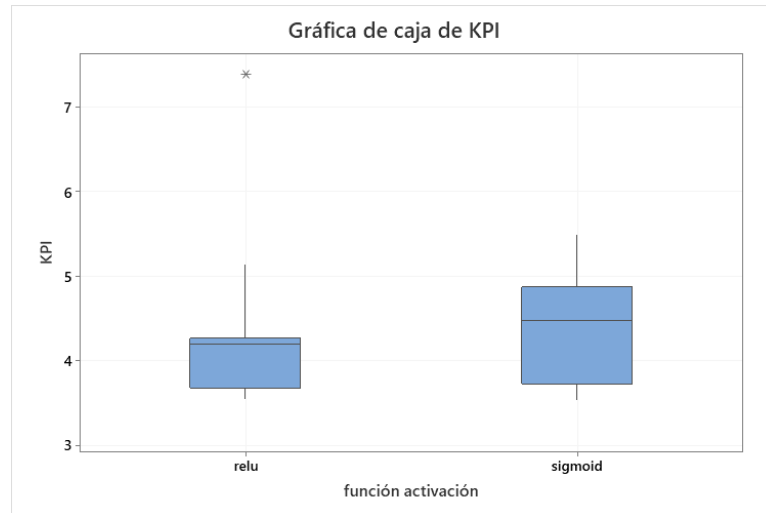


Figura 7.7 Gráfica de caja de los valores de KPI para ambas funciones de activación

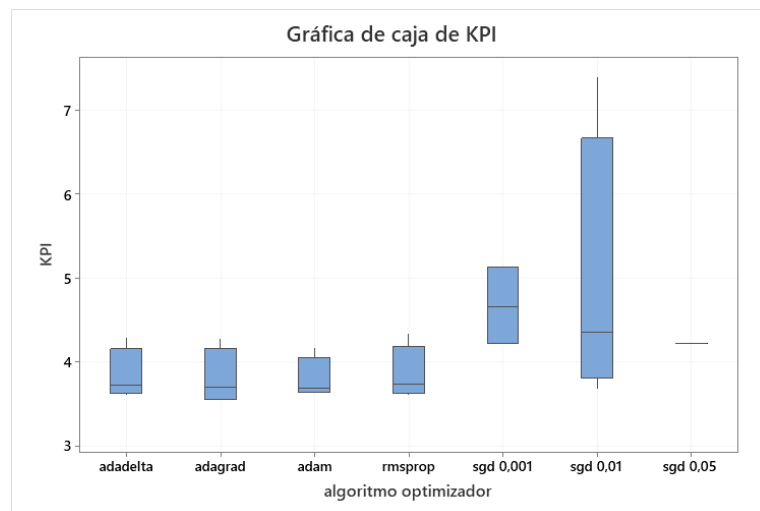


Figura 7.8 Gráfica de caja de los valores de KPI para todos los algoritmos optimizadores

Analizando la **Figura 7.6** se vio como la función de activación ReLu proporcionaba a los modelos una ligera mejoría respecto a la función sigmoid, ya que la mayoría de pruebas ejecutadas que utilizaban ReLu se encontraban en rangos de valores más pequeños del KPI. Además, se pudo confirmar analizando la **Figura 7.7**, donde claramente se veía la tendencia minimalista de las pruebas ejecutadas con la función de activación ReLu.

Para la relación entre el valor de KPI y el algoritmo optimizador, se generó la **Figura 7.8**. En esta se pudo descartar el uso de SGD, debido a su variabilidad y alta media del valor de KPI comparada con los otros optimizadores. Además, en el SGD con ratio de aprendizaje 0.05 se

pudo observar como su método de aprendizaje producía sub-ajuste, se pudo confirmar viendo un gráfico de la predicción y los valores reales (ver Figura.7.9).

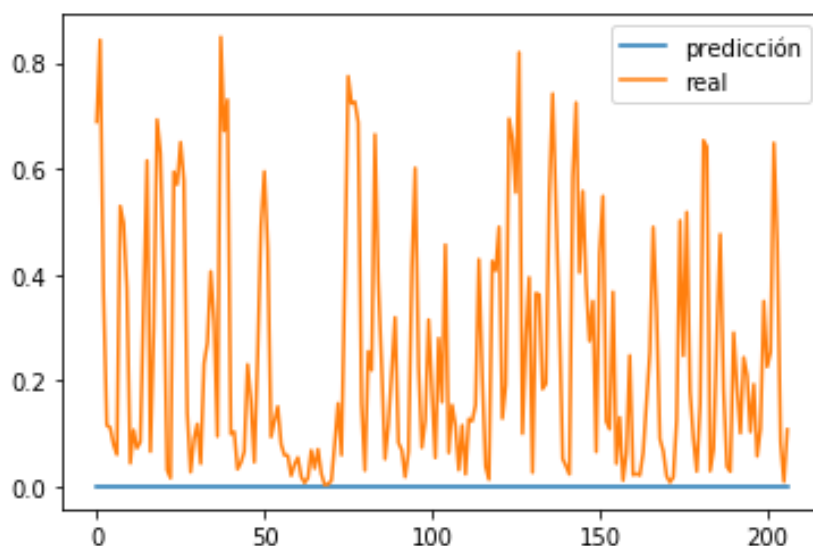


Figura 7.9 Serie temporal de la predicción y el valor real de la producción para SGD 0.05

Modelo frecuencial 2

En este segundo análisis también se realizaron 56 pruebas para poder avanzar hacia el punto óptimo eliminando *hiperparámetros* no relevantes. Posteriormente, al igual que para el otro modelo frecuencial, se analizó el valor y la distribución del KPI para cada función de activación y algoritmo optimizador del aprendizaje.

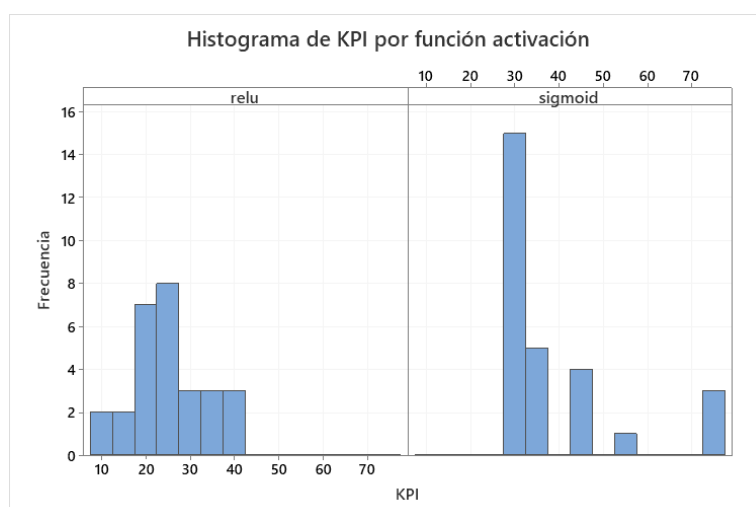


Figura 7.10 Distribución frecuencial de KPI para ambas funciones de activación

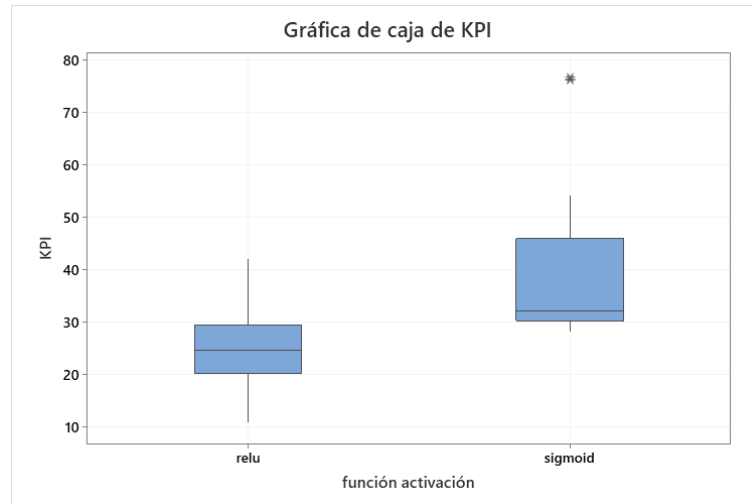


Figura 7.11 Gráfica de caja de los valores KPI para ambas funciones de activación

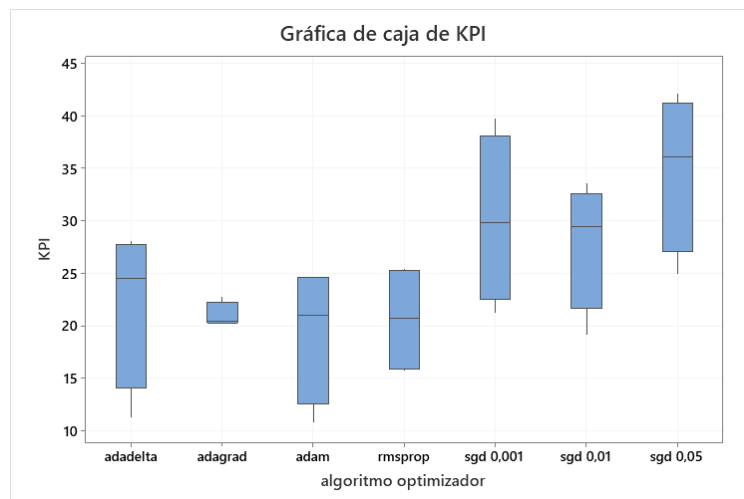


Figura 7.12 Gráfica de caja de los valores de KPI para todos los algoritmos optimizadores

En la primera visión general sobre los datos, se pudo ver como los valores de KPI de todas las pruebas eran bastante más mayores comparados con los del otro modelo frecuencial. Esto es debido a que, al realizar 8 saltos en el tiempo, el error se va acumulando, como ya se había comentado que podía pasar. Aun así, se decidió continuar con el estudio sobre este modelo frecuencial en busca de una mejora.

La relación entre la función de activación y el KPI para este modelo frecuencial se pudo observar en las **Figuras 7.10 y 7.11**. Esta era muy similar a la del anterior, con las mismas tendencias, donde ReLu proporcionaba mejor rendimiento que sigmoid.

Con relación a los algoritmos optimizadores, la **Figura 7.12** mostró como SGD era el que peor rendimiento aportaba. Además, se pudo ver bastante igualdad entre los otros optimizadores. Estos resultados fueron bastante parecidos a los obtenidos por el otro modelo frecuencial.

Para continuar con el test de desarrollo, se decidió escoger de manera definitiva la función de activación ReLu como único *hiperparámetro* para este campo. Además, se decidió prescindir del algoritmo de optimización SGD, con cualquier ratio de aprendizaje, debido a sus malos resultados comparado con el resto. Esto se aplicó a ambos modelos frecuenciales.

7.7.2.2. Paso 2: Sub-modelos

Aprovechando todas las anteriores pruebas, en este segundo paso se decidió observar que sub-modelos dentro de cada modelo frecuencial daban valores de KPI más pequeños (ver Figuras.7.13 y 7.14).

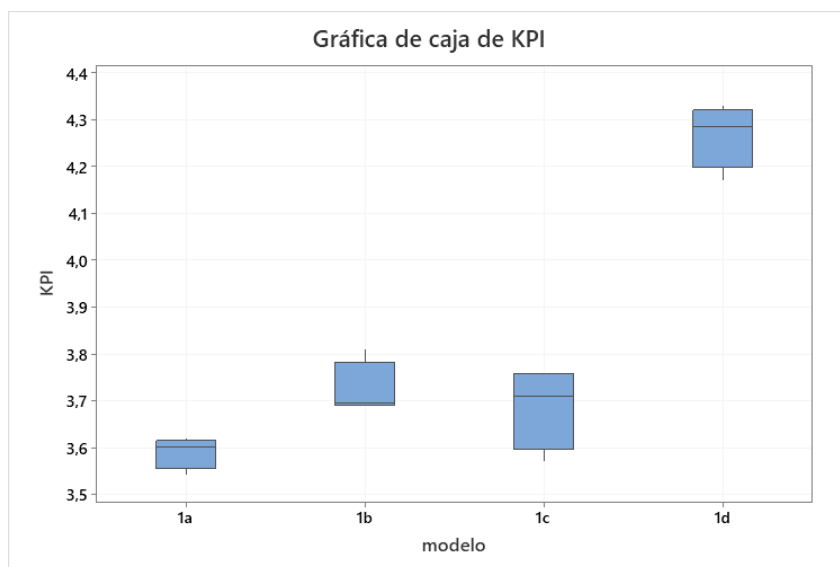


Figura 7.13 Gráfica de caja de los valores de KPI para todos los diferentes sub-modelos frecuenciales 1

Observando la **Figura 7.13**, se pudo ver que el modelo 1d destacaba por su mal rendimiento en comparación con el resto de modelos, por este hecho, se desestimó su uso y se eliminó del test. El modelo 1a, de cuatro entradas, presentaba un comportamiento mejor que el resto, de dos entradas. Aun así, se decidió observar el comportamiento futuro de estos en próximas pruebas, ya que las diferencias eran mínimas.

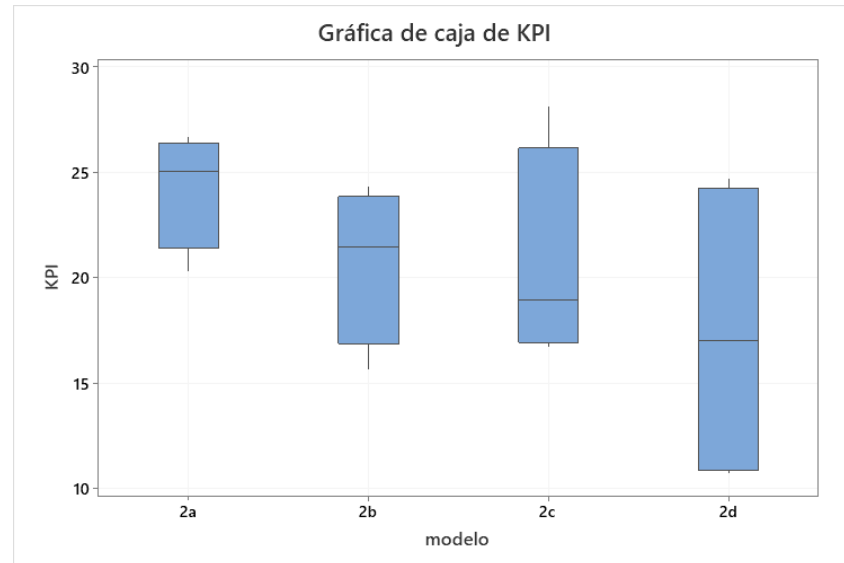


Figura 7.14 Gráfica de caja de los valores de KPI para todos los diferentes sub-modelos frecuenciales 2

Para el segundo tipo de sub-modelos frecuenciales, en la **Figura 7.14**, destacaba el rendimiento del modelo 2d. Este presentaba la media mínima, así como los valores mínimos absolutos de este modelo. En este segundo tipo de sub-modelos, tanto el modelo 2a como el 2d tenían 4 entradas, aun así, su comportamiento era muy distinto. Debido a la gran diferencia con respecto a los valores de KPI del modelo frecuencial anterior, se decidió escoger el modelo 2d como única opción a estudiar.

7.7.2.3. Paso 3: Número de neuronas de la HL

El tercer paso del test de desarrollo se dirigió hacia la optimización del número de neuronas óptimo de las HL, correspondientes a las de la red LSTM en la arquitectura utilizada. Las pruebas se realizaron sobre las conclusiones extraídas en las partes anteriores del test y los números de neuronas se escogieron en potencias de 2 ascendentes, como se había estipulado en el apartado **7.1**. Además, se intentó aprovechar la realización de nuevas pruebas para poder extraer más conclusiones y visualizar tendencias, por eso mismo se decidió estratificar los datos por algoritmo optimizador y modelo (ver Figuras 7.15, 7.16, 7.17 y 7.18).

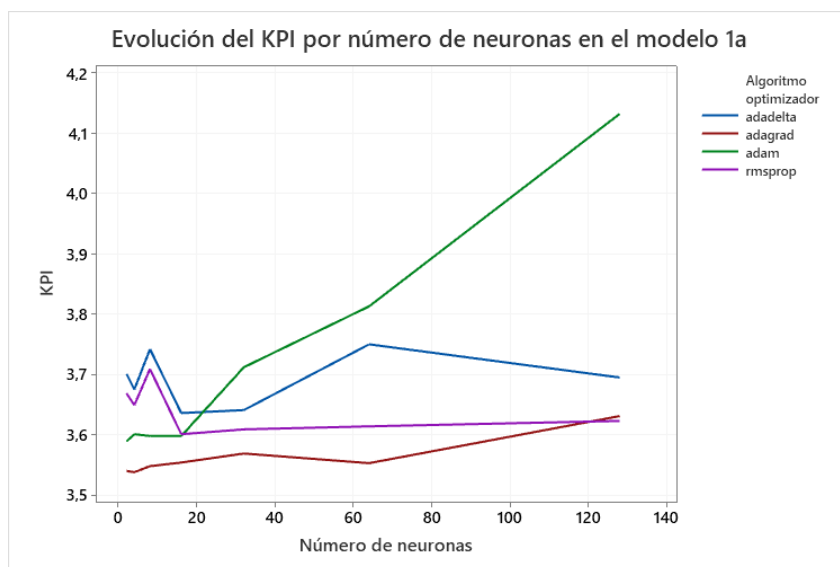


Figura 7.15 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL, para el modelo 1a, estratificado por algoritmo optimizador

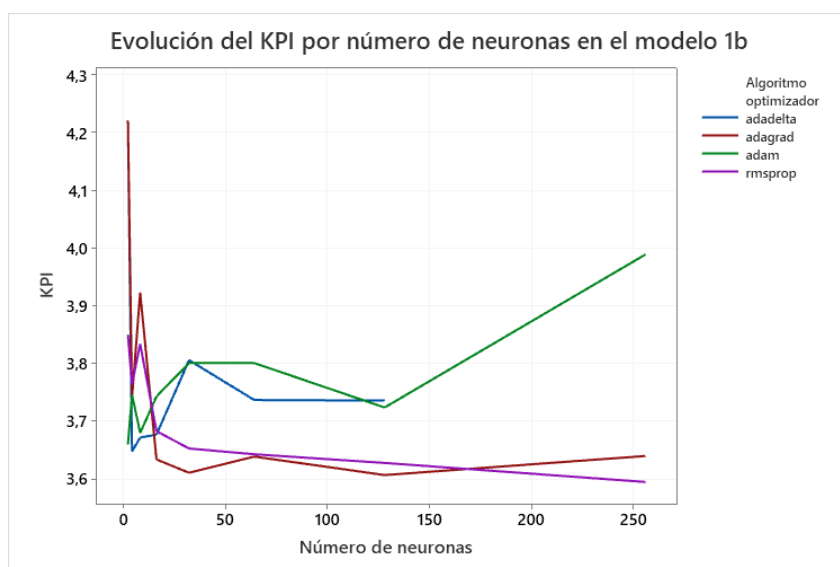


Figura 7.16 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL, para el modelo 1b, estratificado por algoritmo optimizador

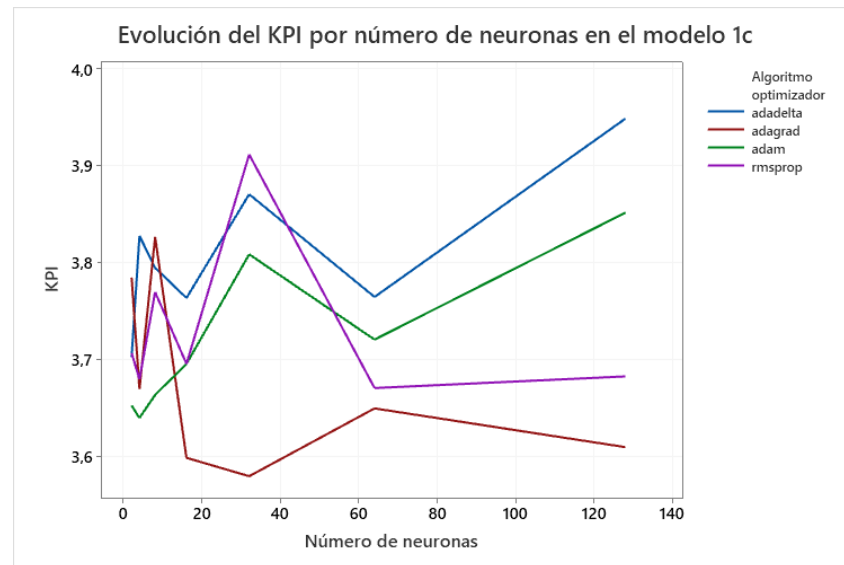


Figura 7.17 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL, para el modelo 1c, estratificado por algoritmo optimizador

En una visión general sobre las **Figuras 7.15, 7.16 y 7.17** se pudo ver como no existía una tendencia clara sobre la cantidad de neuronas con las que se debía construir la HL. Aun así, se disponía de información muy interesante de la cual se extrajeron conclusiones para continuar el análisis.

Analizando de nuevo la **Figura 7.13** a la vez que las mencionadas anteriormente, se pudo ver como los modelos 1a y 1c consiguen valores mínimos de KPI con más normalidad. Esto indica que las variables de entrada común, producción energética total en las últimas 12 horas y velocidad del viento media en las últimas 12 horas, son las más influyentes, como ya indicaba el estudio de correlación en la sección 6.5. Por este mismo motivo, se decidió prescindir del modelo y centrar el trabajo en los modelos 1a y 1c.

En relación con la estratificación por algoritmo de optimización, se pudo ver en las **Figuras 7.15, 7.16 y 7.17** como *Adagrad* minimizaba en todos los casos el valor de KPI. Así, este se escogió como el *hiperparámetro* óptimo para ese campo. Respecto al principal objetivo de este paso del test, el número de neuronas a escoger se decidió esperar a tener más información en los próximos pasos para escoger el valor óptimo.

Observando la **Figura 7.18** a continuación, referente al modelo 2d, se pudo ver como los valores mínimos de KPI se conseguían para valores pequeños del número de neuronas y para los algoritmos optimizadores *RMSprop* y *Adam*. Para el siguiente paso del test de desarrollo se utilizaron los valores 2, 4 y 8 para el número de neuronas y los optimizadores mencionados anteriormente.

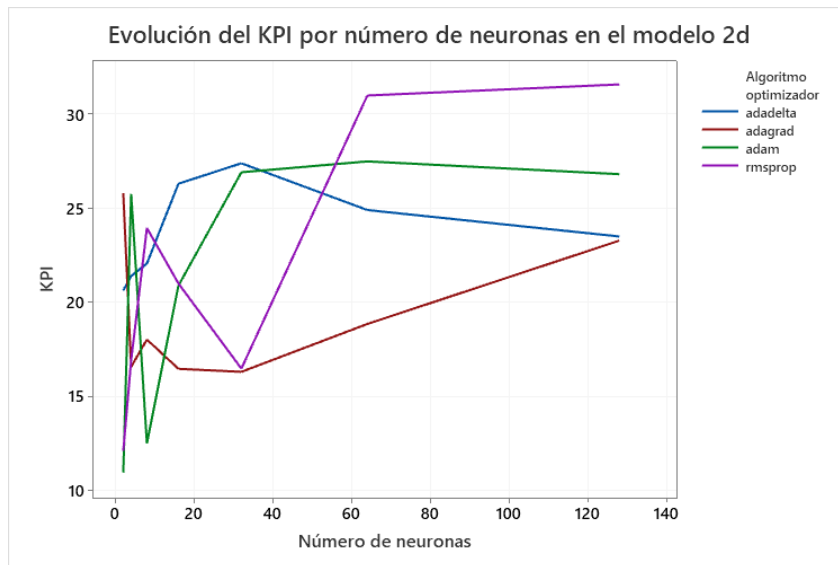


Figura 7.18 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL, para el modelo 2d, estratificada por algoritmo optimizador

7.7.2.4. Paso 4: Medida del lote

Como penúltimo paso de este test de desarrollo, se decidió encontrar el valor óptimo de la medida del lote. Hay que recordar que los candidatos a óptimo son: 1, 16, 32 y 64. En esta parte se tuvo en cuenta el número de neuronas también, para poder conseguir obtener más información sobre este *hiperparámetro*. De esta manera, se realizó una prueba para cada valor posible de número de neuronas para todas las posibles medidas de lote.

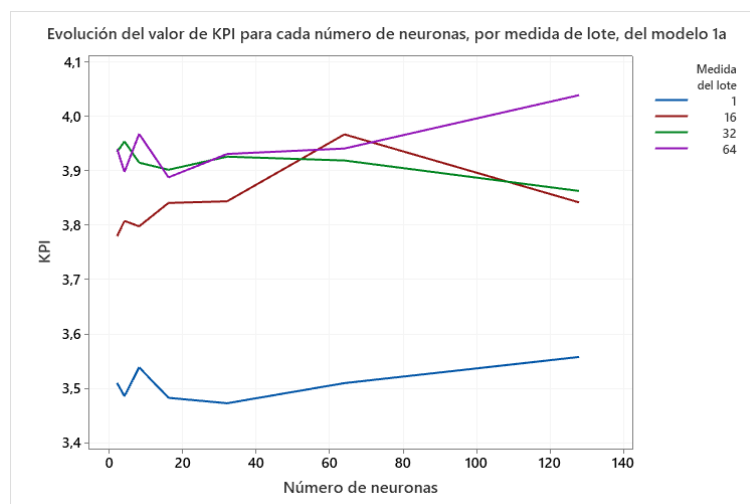


Figura 7.19 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL para el modelo 1a, agrupado por medida de lote

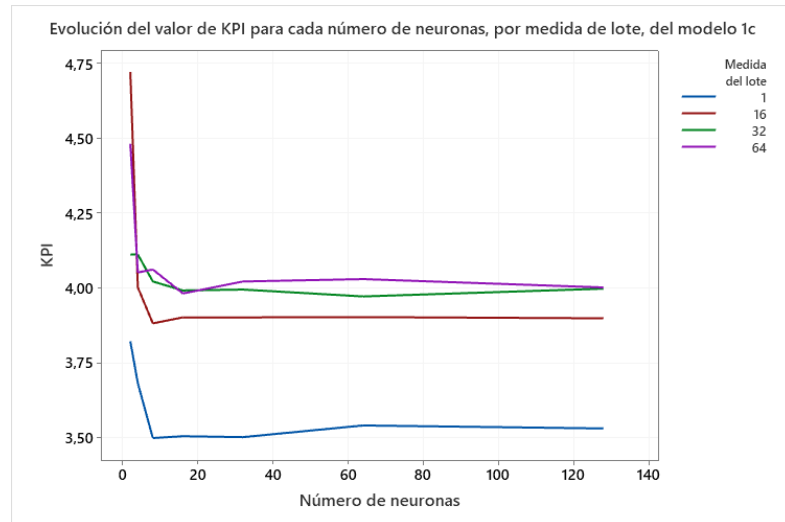


Figura 7.20 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL para el modelo 1c, agrupado por medida de lote

Observando las **Figuras 7.19 y 7.20**, se pudo ver como la medida de lote que minimizaba con diferencia el valor de KPI era de una única muestra o de aprendizaje en línea. Asimismo, se pudo ver como el número de neuronas seguía sin influir de una manera importante en el modelo como para determinar un valor óptimo específico.

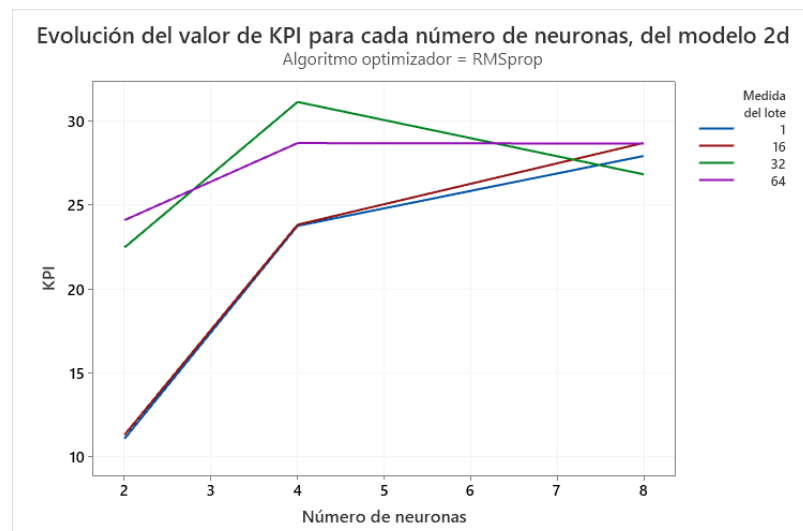


Figura 7.21 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL para el modelo 2d, agrupado por medida de lote y para el optimizador Adam

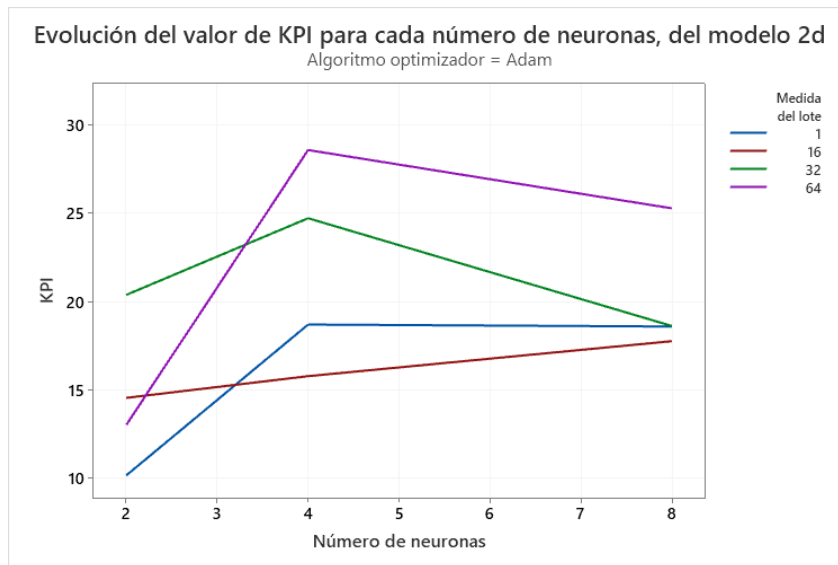


Figura 7.22 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL para el modelo 2d, agrupado por medida de lote y para el optimizador *RMSprop*

Los resultados obtenidos sobre este paso para el modelo 2d se plasmaron en las **Figuras 7.21** y **7.22**. Analizando estas, se vio como la progresión no iba más allá de los mejores resultados obtenidos en los pasos anteriores para valores de KPI.

En acabar este paso, para los modelos 1a y 1c, se decidió escoger la medida de lote de una muestra y no decantarse por ningún modelo ya que ambos predecían de manera similar. En cambio, se decidió descartar definitivamente el modelo 2d debido a que no se veía progresión alguna.

7.7.2.5. Paso 5: Número de épocas

En este último paso, se buscó el valor óptimo del último *hiperparámetro* estructural del modelo, el número de épocas. El número de épocas o iteraciones tenía como objetivo encontrar el punto óptimo de aprendizaje, como se explicó en la sección **5.3.4**, sin sufrir ni sub-ajuste ni sobreajuste. Por lo tanto, se realizaron pruebas sobre ambos modelos restantes en función del número de neuronas de la HL, y los diferentes números de época preseleccionados, parando en el momento que se observaba sobreajuste.

Para seleccionar uno de los dos modelos restantes, se generó la **Figura 7.23** con los valores del análisis. En esta se pudo observar que los mínimos absolutos los conseguía el modelo de

cuatro entradas (1a). Por tanto, se decidió escoger este como el modelo a entrenar.

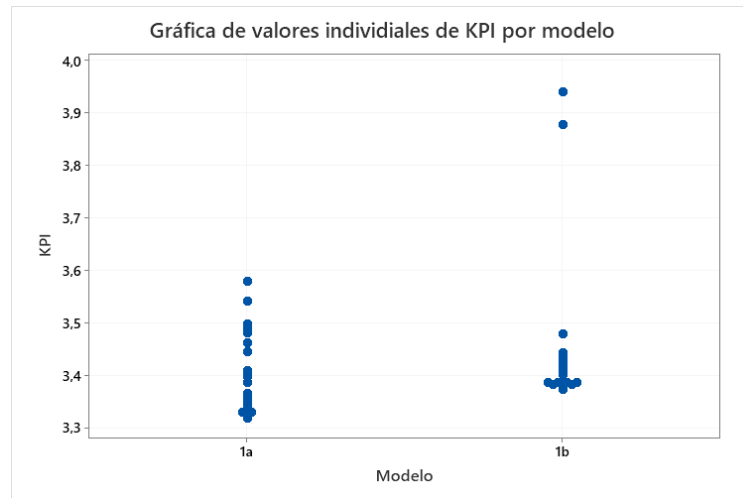


Figura 7.23 Gráfica de valores individuales de KPI para las pruebas realizadas en el paso 5, separadas por modelo

En referencia a los *hiperparámetros* restantes por definir comentados anteriormente, se utilizó la **Figura 7.24**, correspondiente únicamente al modelo 1a, para extraer conclusiones. En esta, se pudo ver que cuando el número de neuronas de la HL era de 8 y 16, se minimizaba el valor de KPI en las pruebas donde el número de épocas era superior o igual a 50. Se decidió escoger los valores de 8 y 50 respectivamente, ya que generaban una red más sencilla de computar. Para valores superiores en el número de neuronas, se producía el efecto no deseado del sobreajuste, mencionado en el apartado 7.5 (ver ejemplo Figura.7.25).

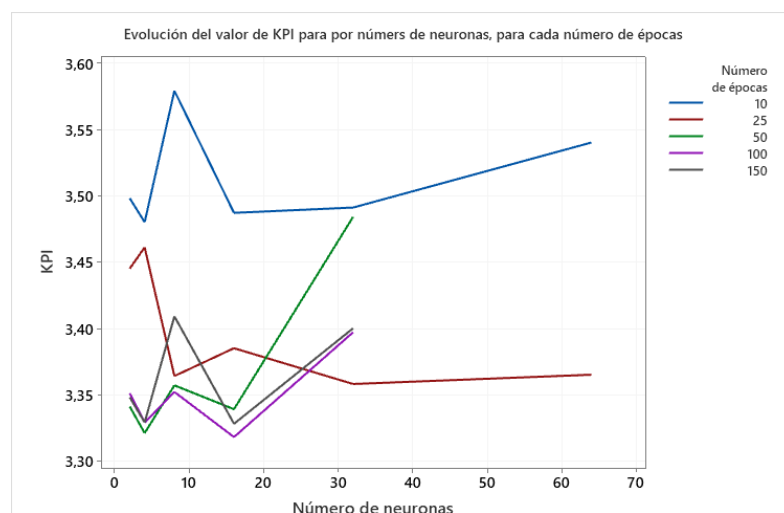


Figura 7.24 Gráfica de la evolución del valor de KPI en función del número de neuronas de la HL para el modelo 1a, agrupado por número de épocas

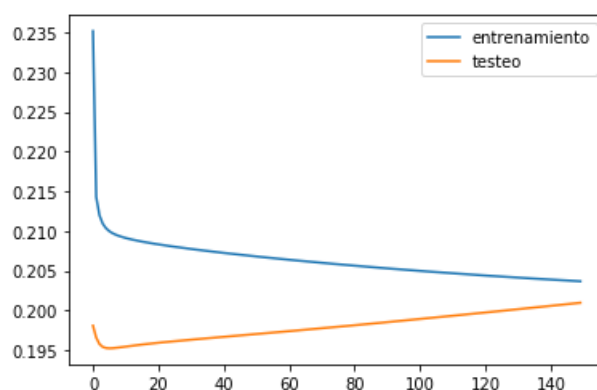


Figura 7.25 Gráfica de la evolución del valor de la función error para ambos grupos de datos durante una prueba de 150 épocas

7.7.3. Resultados

Después de realizar un total de 378 pruebas sobre los distintos modelos propuestos, alternando y comparando diferentes combinaciones de *hiperparámetros* hasta conseguir la que minimizara el valor KPI, se llegó a un conjunto de estos que se resumirá a continuación (ver Tabla.7.1).

<i>Hiperparámetro</i>	<i>Valor</i>
Función de activación de HL	ReLu
Modelo	1a
Número de neuronas de la HL	8
Medida del lote	1
Número de épocas	50
Algoritmo optimizador	Adagrad

Tabla 7.1 *Hiperparámetros* seleccionados para el entrenamiento

Los *hiperparámetros* presentes en la **Tabla 7.1** se utilizarían para el entrenamiento de la red.

8. El modelo predictivo

En este capítulo se expone el procedimiento relacionado con el entrenamiento de la RNA y la creación de predicciones. Comenzando con el entrenamiento de la red definida en la sección anterior y acabando con el análisis de las predicciones generadas por esta desde el punto de vista energético y económico.

8.1. Entrenamiento de la red

Durante el entrenamiento de la red se tuvieron en cuenta todos los *hiperparámetros* encontrados en el test de desarrollo, los cuales permitían generar la RNA más precisa. Siguiendo estas pautas, se dirigió y analizó el entrenamiento de la red hasta crear el modelo predictivo. En el *Anexo B* se encuentra el código con el modelo al completo.

8.1.1. División en entrenamiento y validación

Por norma general, se requiere una división del conjunto de muestras total en entrenamiento y validación para un modelo predictivo basado en una RNA, siempre siendo la dimensión del conjunto de entrenamiento mayor que el de validación. Ambos conjuntos de datos tuvieron para este proyecto la tipología del aprendizaje supervisado, explicada en el apartado 7.6.3, con las características específicas para el sub-modelo frecuencial escogido durante el test de desarrollo (ver Figura.8.1).

Fecha	E12hprevias	V12hprevias	E24hprevias	V24hprevias	Energía
2016-01-03 00:00:00+00:00	12450	2.16806	6.77062e+06	3.94028	16959362
2016-01-04 00:00:00+00:00	1.67141e+07	9.32639	1.69594e+07	5.90208	39174390
2016-01-05 00:00:00+00:00	2.00933e+07	10.3722	3.91744e+07	10.2153	29104329
2016-01-06 00:00:00+00:00	1.21061e+07	8.16667	2.91043e+07	9.20833	2242971
2016-01-07 00:00:00+00:00	1.92806e+06	4.30972	2.24297e+06	3.41111	4371516

Figura 8.1 Primeras 5 muestras del conjunto de datos de entrenamiento del modelo 1a

El conjunto de validación debía seguir dos condiciones: en primer lugar, ser lo suficientemente grande como para producir resultados estadísticamente significativos; en segundo lugar, ser representativo de conjunto de datos en su conjunto [17]. En otras palabras, este conjunto de datos no debía contener diferentes características que el conjunto de entrenamiento.

El primer problema aquí fue la división de los datos en ambos conjuntos. Aunque no hay una solución genérica para este problema, varios factores como las características del problema, el tipo de datos y el tamaño del conjunto de datos se consideraron al tomar la decisión. En general, la dimensión del marco de datos está estrechamente relacionada con la precisión del problema. Cuanta más cantidad de datos, más precisos serán los resultados. Para este problema en particular, el tamaño de los datos fue un factor limitante de la precisión lograda. Dos años de datos en el modelo frecuencial escogido se redujeron únicamente a una muestra por día, con un total de 697, después de las modificaciones realizadas durante el EDA. Por lo tanto, se decidió analizar el valor de KPI en el rango de porcentajes comúnmente utilizados para este tipo de problemas (ver Figura.8.2).

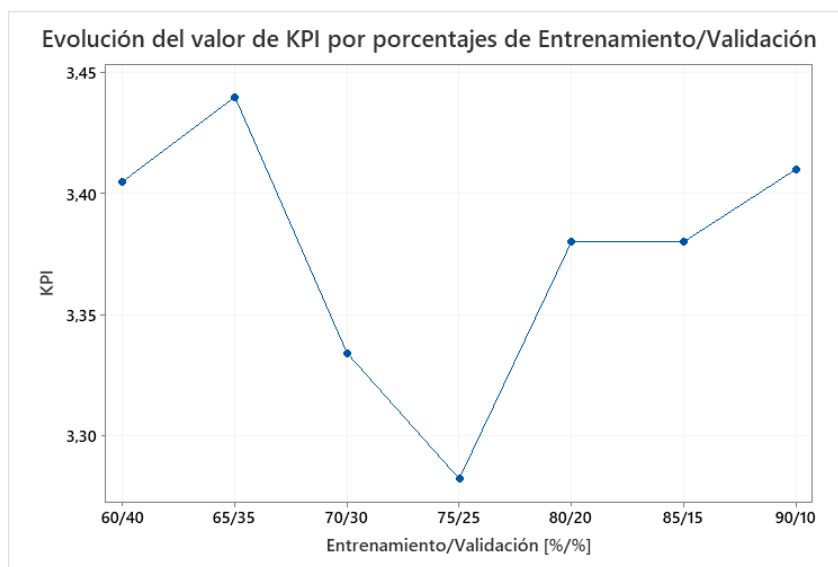


Figura 8.2 Gráfica del valor de KPI en función de la división porcentual de los datos en entrenamiento y validación

El análisis realizado sobre los porcentajes de la división en entrenamiento y validación se graficó en la **Figura 8.2**. En esta, se observó como el punto óptimo se logró con un 75% de las muestras totales para el entrenamiento y el 25% restante para la validación. Esta división era capaz de proporcionar datos estadísticamente significativos, como se había mencionado anteriormente que era necesario. Se utilizaron 522 datos o días en el entrenamiento y 175 en la validación.

En el *Anexo B*, dentro del código completo del modelo se encuentra el código utilizado para esta división.

8.1.2. Ejecución del entrenamiento y resultados obtenidos

Después de dividir los datos en dos y haber preparado la red para el entrenamiento y la validación correspondiente, el algoritmo fue compilado. Para ello se utilizaron las bibliotecas *keras* y *tensorflow* de *Python*, ya mencionadas en la sección 7.1.

Durante todo el entrenamiento, los datos se encontraron normalizados, es decir, en valores entre 0 y 1. Esto, sin embargo, permitía de la misma manera monitorizar el error después de cada época, aunque en unidades normalizadas en el caso del RMSE y el MAE. Además, las bibliotecas mencionadas anteriormente permitieron un seguimiento del valor de la función error y las métricas sobre los datos de entrenamiento y validación al final de cada época.

Una vez finalizado el entrenamiento, con una duración total de 150 segundos, 3 por época, se pasó a graficar los diferentes resultados para obtener una visión general sobre lo que podría ser la predicción. Para observar de una manera más óptima estos resultados se decidió reajustar el rango a los valores originales, deshaciendo la normalización.

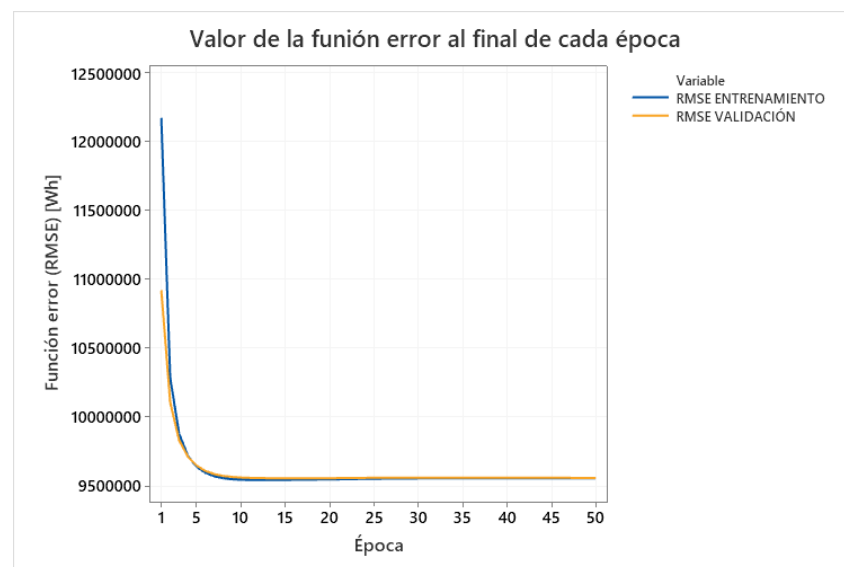


Figura 8.3 Gráfica de la evolución del valor de la función error (RMSE) en Wh sobre los conjuntos de entrenamiento y validación al final de cada época

El resumen del entrenamiento se graficó en la **Figura 8.3**. En esta, se observó como en un principio la red consiguió aprender de manera muy rápida, reduciendo el valor de la función error a grandes saltos al final de cada época, hasta que se estabilizó alrededor de un valor. Sin embargo, se pudo ver como el entrenamiento no sufría ni sobreajuste ni sub-ajuste. La evolución similar de la función error sobre ambos conjuntos se produjo por la inexistencia de estacionalidad en los datos, más allá de la observada en el EDA entre el día y la noche, la

cual se encontraba incorporada en cada muestra diaria, y por lo tanto no influyó.

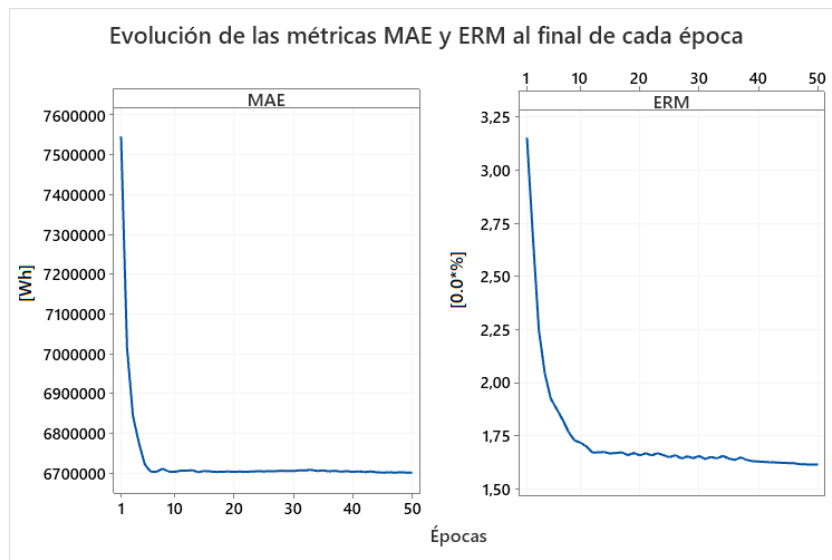


Figura 8.4 Gráfica de la evolución del valor de las métricas (MAE y ERM) sobre el conjunto de validación al final de cada época

Para observar de la misma manera la evolución de las métricas se generó la **Figura 8.4**. En este caso, solo se analizó su evolución sobre el conjunto de validación, ya que se había visto anteriormente que era indiferente. El resultado que se obtuvo fue muy similar al de la **Figura 8.3**, todo el conjunto de función error y métrica estaba en sintonía y sin indicios de sobreajuste, muy importante para obtener resultados fiables. Por último, se decidió graficar la evolución de KPI, donde ambas métricas entraban en relación (ver Figura 8.5).



Figura 8.4 Gráfica de la evolución del valor de KPI al final de cada época

8.2. Análisis de las predicciones

La creación de un modelo predictivo basado en una red LSTM fue el fin de este proyecto, así como las predicciones que este genera, por ello debían ser estudiadas y analizadas en profundidad. En este apartado se analizó la precisión de estos pronósticos y se estimó su impacto en la retribución económica, todo esto permitió exponer la calidad del modelo creado.

8.2.1. Visión general

Este primer análisis se realizó como una visión general sobre los datos ya disponibles, los cuales provenían del apartado anterior. Los datos sobre los cuales se pudo realizar este análisis y sacar resultados objetivos fueron del conjunto de validación, ya que el modelo entrenado no conocía los resultados, lo cual no pasaba con el conjunto de entrenamiento. Entonces, se comparó el conjunto de muestras de validación real con el pronosticado por el modelo para el mismo grupo.

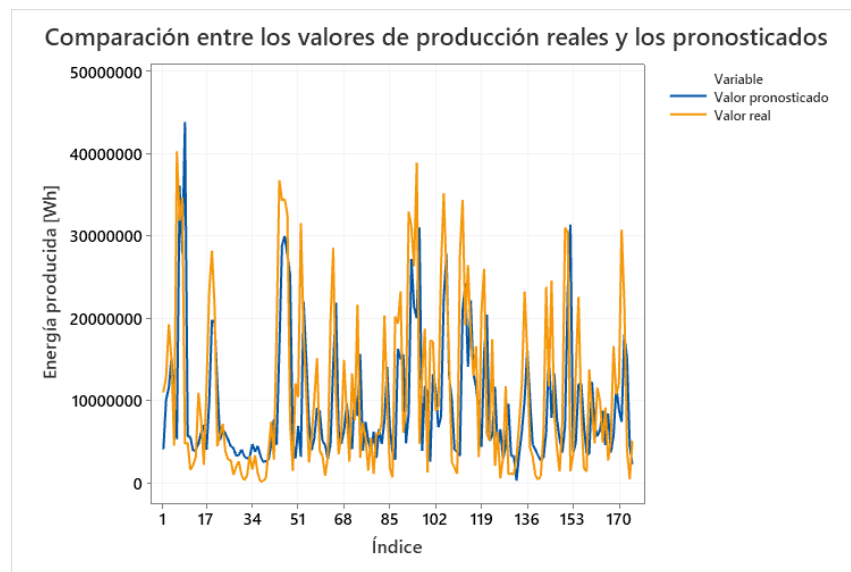


Figura 8.5 Gráfica comparativa de los valores de producción reales y los pronosticados para las 175 muestras diarias del conjunto de validación, en Wh

Para empezar este estudio, se decidió observar la diferencia en las 175 muestras del conjunto. Así, se creó la **Figura 8.5**, donde se vio una gran similitud entre ambos grupos de muestras. Sin embargo, se observó como el modelo tenía una tendencia a predecir un valor similar al del día anterior, es decir, al de las entradas.

A continuación, se exponen dos tablas que se utilizaron para analizar el valor de la función error, las métricas, KPI y algunos datos estadísticos más considerados de interés en la

comparación (ver Tablas.8.1 y 8.2).

Parámetro comparativo	Valor
Función error (RMSE)	9.574.856,244 Wh
Métrica MAE	6.717.453 Wh
Métrica ERM	1,586
KPI	3,267

Tabla 8.1 Datos comparativos y métricas

Datos estadísticos

Variable	N	Media (Wh)	Desv.Est. (Wh)	Suma (Wh)	Mínimo (Wh)	Máximo (Wh)
Predicción	175	9.510.922	7.666.409	1.664.411.271	252.551	43.779.600
Real	175	11.217.519	10.335.522	1.963.065.865	149.866	40.208.697

Tabla 8.2 Descripción estadística de ambos conjuntos de datos

Al observar las **Tablas 8.1 y 8.2** se pudieron extraer diversas conclusiones. Primero, se confirmó que los valores generados por el modelo eran similares a los del día anterior, ya que, observando la variable suma de la segunda tabla, se aprecia como la cantidad de energía producida total es muy similar, cuando las métricas de la primera tabla no dicen lo mismo en la comparación valor a valor. Asimismo, observando que la desviación estándar era sustancialmente menor para el conjunto de datos predichos se intuyó que los valores predichos se encontrarían en un rango de valores más estrecho que los reales. Por último, la tabla permitió ver que todos los valores predichos se encontraban dentro del rango de valores original y físicamente posibles.

Para acabar con esta visión general se analizó lo que era el objetivo de este proyecto, la comparación valor a valor, es decir, la precisión del modelo a 24 horas vista. Comparando la media de la **Tabla 8.2** con el error medio absoluto de la **8.1** se pudo ver como este segundo suponía un 70% del valor del primero. Además, el error medio relativo fue de 1,5 veces el

valor de la predicción. Ambos valores indicaron una alta imprecisión a corto plazo.

8.2.2. Pronosticación de la retribución económica

Para la realización de este apartado de manera detallada, antes que nada, se buscaron los precios medios diarios de la energía para la segunda mitad del año de 2017. Fueron especialmente estas fechas ya que dentro de ese periodo se encontraban los 175 datos diarios del conjunto de validación, también usados para este apartado (ver Figura.8.6).

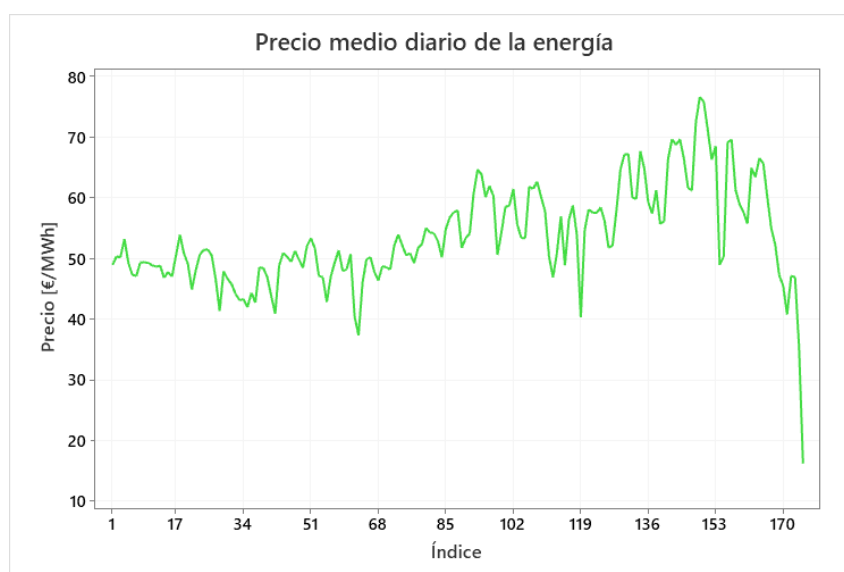


Figura 8.6 Gráfica del precio medio diario de la energía durante la segunda mitad de 2017

Para resumir los resultados de la pronosticación de la retribución económica se utilizaron las mismas tablas que en el apartado anterior, con algunas modificaciones (ver Tablas.8.3 y 8.4). Estas permitieron de nuevo realizar un análisis a corto plazo, el cual era el que interesaba.

Parámetro comparativo	Valor
Métrica MAE	363,48€
Métrica ERM	1,586

Tabla 8.3 Métricas comparativas

Datos estadísticos

Variable	N	Media (€)	Desv.Est. (€)	Suma (€)	Mínimo (€)	Máximo (€)
Retribución real	175	605,1	571,0	105.894,8	6,4	2.343,9
Retribución pronosticada	175	513,6	424,8	89.874,7	15,1	2.158,3

Tabla 8.4 Descripción estadística de ambos conjuntos de datos

Analizando las **Tablas 8.3** y **8.4** se pudieron ver las mismas características en ambos conjuntos de datos que en el apartado anterior, pero en unidades monetarias y para los precios de la energía de ese periodo de tiempo. Aun así, se observó una diferencia en el máximo provocada por las variaciones del precio a lo largo del periodo temporal.

Por último, se hizo un breve análisis del impacto a largo plazo generando la **Figura 8.7**, esta permitió ver los beneficios económicos acumulados a medida que se avanzaba en el tiempo. Así, se observó que a largo plazo la eficacia del pronosticador era mucho mayor, también en parte porque el modelo pronostica valores muy parecidos a los del día anterior, como ya se había visto.

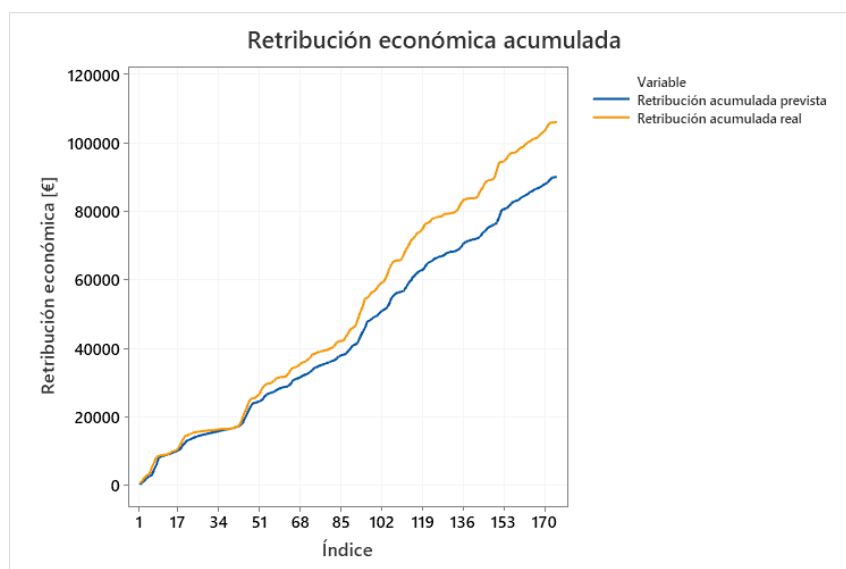


Figura 8.6 Gráfica de la retribución económica acumulada para ambos conjuntos

8.2.3. Valoración de las predicciones

Tanto para la predicción energética como para la predicción económica los errores fueron muy significativos. En general, el modelo creado era incapaz de detectar con anterioridad cambios en la producción a un día vista y se limitaba en la mayoría de los casos a reproducir un valor similar al del día anterior. De esta manera, la comparación de la suma de energía y de dinero total era muy similar para los conjuntos de datos, pero en la comparación diaria valor a valor no se reflejaba lo mismo.

Entonces, se decidió graficar la métrica del error relativo para poder mapear su evolución a lo largo de las 175 muestras de datos. (ver Figura.8.7).

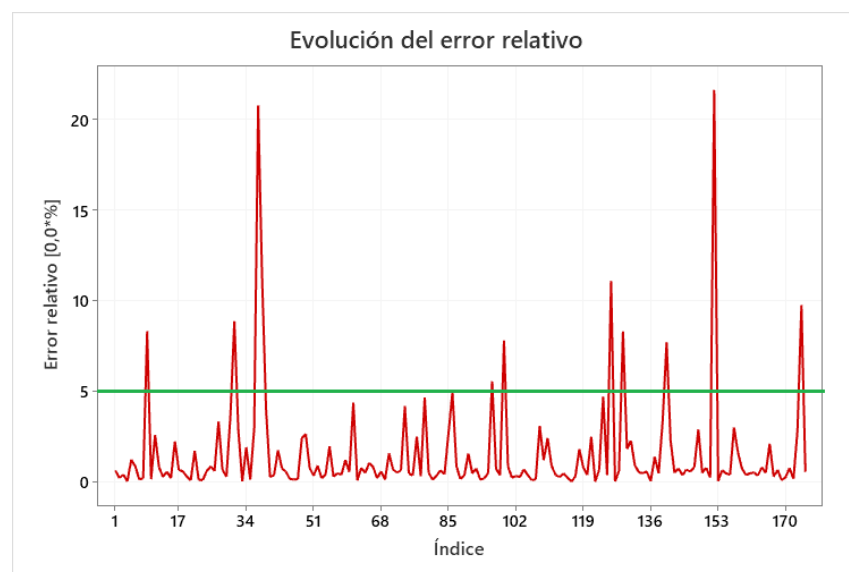


Figura 8.7 Gráfica de la evolución del error relativo entre los valores previstos y los reales

En observar la **Figura 8.7** se pudo ver como las predicciones no tenían tan mala precisión como se había supuesto hasta ahora. En esta gráfica se vio como en la mayoría de días de este conjunto, el error se encontraba por debajo de 5 veces el valor de la producción, límite indicado por la línea verde. De hecho, solo un total de 11 muestras del conjunto pronosticado se encontraban por encima de ese límite, lo que suponía un 6,3% del total. Sin embargo, estas 10 muestras en las cuales el valor pronosticado difería en más de 5 veces el valor real representaban una variación del valor del ERM del 42%, es decir, pasaba de un valor de 1,586 a 0,924.

Después de observar que unas pocas muestras del ERM generaban un cambio tan grande en las métricas y por lo tanto influían en la precisión del modelo de una manera tan importante, se decidió observar algunos esos puntos críticos de una manera detallada (ver Figuras.8.8 y

8.9).

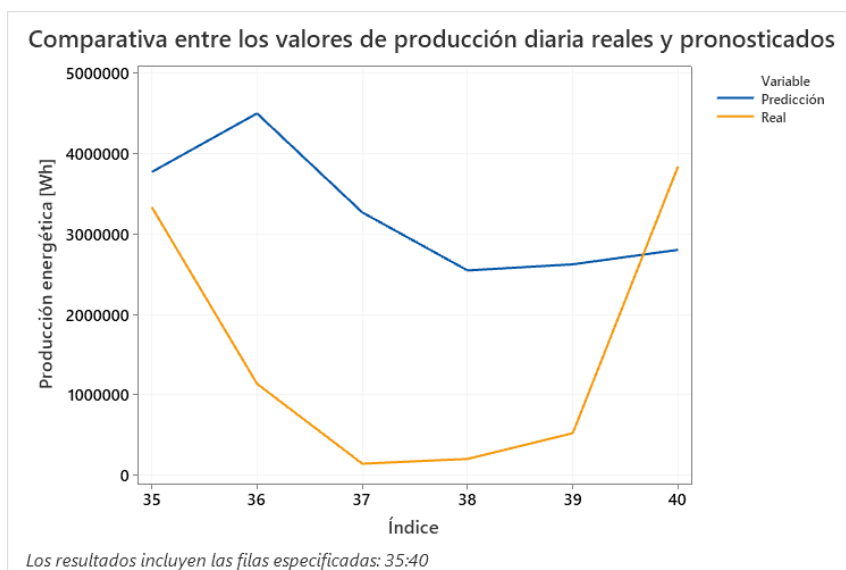


Figura 8.8 Gráfica comparativa entre los valores de producción diaria reales y pronosticados en el rango de índices 35-40

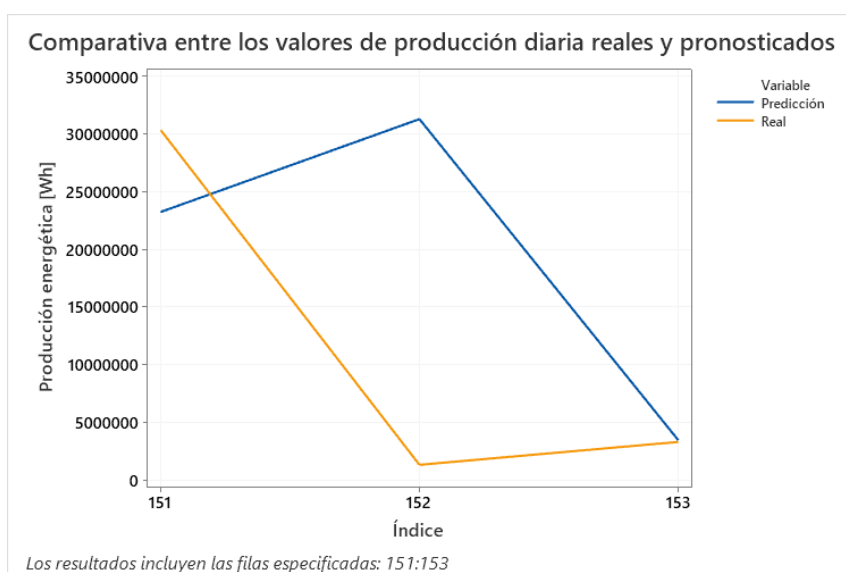


Figura 8.9 Gráfica comparativa entre los valores de producción diaria reales y pronosticados en el rango de índices 151-153

En las **Figuras 8.8 y 8.9**, las cuales pertenecían a los dos valores de error relativo mayor de entre todo el conjunto de datos, se pudo ver el mismo suceso. En ambos casos el valor real era muy pequeño, casi 0, y eso provocaba que, aunque el valor del error no fuera grande, en relación con el real se disparaba.

Respecto a la métrica observada por el MAE, el hecho de que fuera un 70% de la media implicaba una gran imprecisión del modelo. Pero observando que los valores reales se encontraban entre un mínimo en 149.866 y un máximo de 40.208.697, se llegó a la conclusión que un MAE de 6.717.453 no significaba tanto error como aparentó en un principio. Y menos teniendo en cuenta la inexistencia de una estacionalidad y la gran variabilidad del viento. Todos los valores numéricos mencionados en este párrafo provienen de las **Tablas 8.1 y 8.2**.

Los comentarios realizados sobre la predicción en este apartado son extrapolables a los pronósticos de retribución económica ya que el cálculo de esta es el producto del precio por la producción energética.

9. Análisis de costes

El coste total asociado al proyecto fue la combinación del coste del trabajo del autor y el coste del equipo, software más hardware. Todo el trabajo se realizó con las diferentes herramientas necesarias para sacar el máximo rendimiento a los datos y representar los resultados de la manera más clara y concisa posible.

9.1. Coste del trabajo humano

Durante 3 meses, el estudiante ha estado trabajando en la tesis. Los dos primeros meses, durante 5h/día, realizando la investigación, análisis y cálculo necesarios; y el mes restante, un promedio de 8h/día, realizando la redacción y diseño de la memoria. Además de algunas horas extras para las reuniones con el tutor y últimos detalles. Concluyendo en un total de 400h de trabajo. Trabajar como un científico de datos profesional estipuló un precio de 25 €/h por el tiempo de trabajo.

Tarea	Tiempo [h]	Coste [€]
Investigación	140	3.500
Programación y análisis	100	2.500
Redacción	160	4.000
Total	400	10.000

Tabla 8.3 Coste autor

9.2. Coste del hardware y software

El hardware utilizado fue un ordenador MSI GP62MVR 6RF Leopard Pro. Este se utilizó en todo momento, para la investigación, programación, análisis y redacción del proyecto. El precio de este dispositivo es de 1.034,75€, con una vida útil de 7 años, tiene un coste de computado de 12,32€/mes. Esto supuso un aumento en el coste del proyecto de 36,96€.

En lo que al software se refiere, se utilizó el siguiente programario:

- *Spyder*, un entorno de desarrollo integrado multiplataforma de código abierto para programación científica en lenguaje *Python*. Por lo tanto, su coste fue de 0€.
- Para el análisis estadístico, graficación y la gestión de datos se utilizó *Microsoft Office Excel* y *Minitab*. Con unos respectivos costes de 149€ correspondiente al *Pack Office Hogar*, que incluye *Excel*, y 29,99€ para *Minitab*.
- En la redacción y edición de la memoria se utilizó *Microsoft Office Word*, incluido en el *Pack Office Hogar*.

9.3. Coste total

El coste total del proyecto ha sido de 10.215,95€.

10. Impacto ambiental

La realización del proyecto en sí no tuvo impacto positivo ni negativo en el medio ambiente. Sin embargo, la implementación de un pronóstico preciso de la producción de energía eléctrica a corto plazo de un aerogenerador puede aumentar el rendimiento de la red de distribución. Esta tecnología permite una anticipación en la distribución de la energía y, por tanto, una reducción en la pérdida de energía y en el coste de la distribución. En consecuencia, causaría un efecto positivo en el medio ambiente.

Conclusiones

La utilización de la última tecnología en redes neuronales es un campo fundamental a tener en cuenta para extraer el máximo aprovechamiento a la gran cantidad de datos que se almacenan diariamente a nivel mundial. Esto no es únicamente aplicable a nivel de la energía eólica o aerogeneradores, sino en todos los ámbitos, casi cualquier cosa que pasa en el mundo podría ser pronosticada con anterioridad utilizando redes neuronales.

En el proyecto realizado se describe todo el proceso de creación de un modelo predictivo basado en una red neuronal LSTM, centrado en la pronosticación de la producción de energía eléctrica de un único aerogenerador para las siguientes 24 horas. Todo ello consiguiendo la mayor precisión posible en el modelo final.

Dicho proyecto empieza con un análisis exploratorio de datos sobre el marco de muestras disponible. Además de crear dos modelos frecuenciales, este análisis logró filtrar y seleccionar los datos para sacar el máximo rendimiento de estos en pasos futuros. Sobre los datos seleccionados, se realizaron dos acciones:

- Buscar correlaciones fuertes entre características pasadas de los datos y futuras, para poder aumentar la precisión del modelo en su utilización como entradas.
- Analizar la existencia de estacionalidades, a corto o largo plazo, que pudieran influir en la exactitud del modelo.

A continuación, se realizó el proceso de construcción, donde el objetivo principal era la selección de los *hiperparámetros* óptimos para este específico modelo pronosticado. En esta etapa, se deben destacar dos acciones:

- Crear sub-modelos frecuenciales con diferentes características y números de entradas entre los cuales se escogería el de más precisión.
- Realizar un test de desarrollo que expusiera todos los *hiperparámetros* óptimos del modelo, entre ellos el sub-modelo frecuencial.

Por último, el modelo predictivo fue expuesto a entrenamiento y al análisis de sus pronósticos. En esta última etapa del proyecto, además de analizar en profundidad la precisión del modelo, se analizó su aplicación económica.

Realizando el EDA, se apreció la importancia de seleccionar los datos con los que entrenar el modelo de manera precisa. Además, en este mismo apartado se predijo la dificultad que iba a tener el futuro modelo para predecir con exactitud, debido a la falta de estacionalidad observada y gran variabilidad de los datos.

Entrando en la construcción del modelo, la amplia gama de posibles valores para cada *hiperparámetro* puso de manifiesto la importancia de realizar una búsqueda robusta para cada uno de ellos. Así como la importancia de crear un KPI que generalice diferentes aspectos a tener en cuenta en un solo valor, reduciendo así una gran cantidad de trabajo.

En cuanto al acto de pronosticar, se pudo ver cómo una red LSTM con pocas entradas y de rápido entrenamiento fue capaz de predecir valores futuros con un error aceptable únicamente utilizando datos de experiencias previas. Aceptable teniendo en cuenta que se trata de un único aerogenerador y que los datos de origen presentan estacionalidad nula y alta varianza. Aún así, se vio como el pronosticador era incapaz en la mayoría de los casos de prever grandes variaciones en la producción de energía y se limitaba a reproducir un valor similar al del día anterior en la mayoría de los días. Por esta misma razón, se detectó una falsa precisión a largo plazo.

En definitiva, este proyecto ha servido como toma de conciencia de la importancia de una buena utilización de los datos de los que se dispone. Así como del gran potencial que tiene la utilización de la tecnología del aprendizaje profundo y en especial las redes LSTM en el ámbito de la pronosticación.

Adicionalmente, este proyecto me ha proporcionado una visión alternativa a la que ya tenía sobre el análisis de datos. Esta breve pero intensa introducción a la ciencia de datos me ha servido para ver que es un campo que crea un gran interés en mí.

Agradecimientos

Quisiera agradecer antes que nada a mis padres por su apoyo incondicional en todo momento, en especial a mi padre, por además aportar su granito de arena en este proyecto. También a mi tutor por ayudarme a buscar un tema adecuado a mis gustos y tutelarme en todo momento.

Por último, quisiera agradecer a *EDP Renováveis*, ya que gracias a compañías como esta que ponen sus datos a disposición de todo el mundo se pueden realizar proyectos como este, que son el futuro de la innovación.

Bibliografía

Referencias bibliográficas

- [1] ASOCIACIÓN EMPRESARIAL EÓLICA (AEE), *La eólica supera los 25.700 MW instalados en España tras un año de intensa actividad*, [<https://www.aeeolica.org/comunicacion/la-actualidad-eolica-en-5-minutos/boletin-febrero-2020/4019-la-eolica-supera-los-25-700-mw-instalados-en-espana-tras-un-ano-de-intensa-actividad>, 25 de febrero de 2020]*.
- [2] JOSÉ MIGUEL VIÑAS RUBIO, *Origen y desarrollos actuales de la predicción meteorológica*, [<http://www.divulgameteo.es/uploads/Origen-desarrollos-predicci%C3%B3n.pdf>, 25 de febrero de 2020]*.
- [3] PERIÓDICO ONLINE ENERGÍAS RENOVABLES, *El mantenimiento predictivo puede ahorrar cinco millones de euros a un parque eólico*, [<https://www.energias-renovables.com/eolica/el-mantenimiento-predictivo-puede-ahorrar-cinco-millones-20181205>, 25 de febrero de 2020]*.
- [4] OMIE, *Nuestros mercados de electricidad*, [<http://m.mibgas.es/inicio/mercados-y-productos/mercado-electricidad/nuestros-mercados-de-electricidad/>, 17 de febrero de 2020]*.
- [5] COLAH'S BLOG, *Understanding LSTM Networks*, [<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>, 10 de marzo de 2020]*.
- [6] JASON BROWNLEE, *A Gentle Introduction to Mini-Batch Gradient Descent and How to Configure Batch Size*, [<https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descent-configure-batch-size/>, 2 de abril de 2020]*.
- [7] JASON BROWNLEE, *Difference Between a Batch and an Epoch in a Neural Network*, [<https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/>, 2 de abril de 2020]*.
- [8] OUR WORLD DATA, *Carbon dioxide (CO₂) emissions by sector or source, World, 1960, to 2014*, [<https://ourworldindata.org/grapher/carbon-dioxide-co2-emissions-by-sector-or-source>, 10 de abril de 2020]*.
- [9] MICHAEL Y. HU GUOQIANG ZHANG B. EDDY PATUWO, *Forecasting with artificial neural networks: The state of the art*, International Journal of Forecasting (July 2014), p. 35-37.

- [10] JUAN IGNACIO BAGNATO, *Breve Historia de las Redes Neuronales Artificiales* (septiembre, 2018).
- [11] A. GACHAGAN C. E. NWANKPA W. IJOMAH AND S. MARCHALL, *Activation Functions: Comparison of trends in practice and research for deep learning*, [<https://arxiv.org/abs/1811.03378>, 15 de marzo de 2020]*.
- [12] WILL KOEHRSEN, *Overfitting vs. Underfitting: A Conceptual Explanation*, [<https://towardsdatascience.com/overfitting-vs-underfitting-a-conceptual-explanation-d94ee20ca7f9>, 15 de marzo de 2020]*.
- [13] KAMAL MALIK, HARSH SADAWARTI AND GURSHARANJEET KALRA, *Comparative Analysis of Outlier Detection Techniques*, Volume 97– No.8, July 2014.
- [14] FARHAD MALIK, *Understanding what really happens in a neural network; Neural Network Activation Function Types*, [<https://medium.com/fintechexplained/neural-network-activation-function-types-a85963035196>, 30 de marzo de 2020]*.
- [15] JASON BROWNLEE, *How to Configure the Number of Layers and Nodes in a Neural Network*, [<https://machinelearningmastery.com/how-to-configure-the-number-of-layers-and-nodes-in-a-neural-network/>, 30 de marzo de 2020]*.
- [16] ROBERT R.F. DEFILIPPI, *Standardize or Normalize? — Examples in Python*, [<https://medium.com/@rrfd/standardize-or-normalize-examples-in-python-e3f174b65dfc>, 1 de abril de 2020]*.
- [17] GOOGLE, *Conjuntos de entrenamiento y prueba: Separación de datos*, [<https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data>, 25 de abril de 2020]*.

Bibliografía complementaria

- [18] EDP RENOVÁVEIS, *Datos usados para el proyecto*, [<https://opendata.edp.com/explore>, 20 de febrero de 2020]*.
- [19] OMIE, *Datos de los precios de la energía*, [<https://www.omie.es/>, 10 mayo de 2020]*.
- [20] KERAS, *Información biblioteca Keras*, [<https://keras.io/>, 15 de marzo de 2020]*.